

Ein Kommunikations- und Tutoring-System für Lerngruppen im Internet

Inauguraldissertation
zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
der Universität Mannheim

vorgelegt von

Diplom-Ingenieur
Hans Christian Liebig
aus Biedenkopf

Mannheim, 2005

Dekan: Professor Dr. Matthias Krause, Universität Mannheim
Referent: Professor Dr. Wolfgang Effelsberg, Universität Mannheim
Korreferent: Professor Dr. Colin Atkinson, Universität Mannheim

Tag der mündlichen Prüfung: 05. August 2005

Vorwort

Die vorliegende Arbeit entstand im Rahmen meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Praktische Informatik IV der Universität Mannheim. Mein besonderer Dank gilt dem Betreuer dieser Arbeit, Herrn Professor Dr.-Ing. Wolfgang Effelsberg, er gab mir die Freiheit einer eigenen wissenschaftlichen Entwicklung, begleitete mich hierbei kritisch wie auch konstruktiv. Insbesondere ermöglichte er mir die Teilnahme an Konferenzen und dadurch die Integration in die wissenschaftliche Fachwelt.

Beim meinem Zweitgutachter Herrn Professor Dr. Colin Atkinson möchte ich mich ebenfalls bedanken.

Für die Korrektur der Arbeit danke ich ganz herzlich Ulrike Moroff und Joachim Wirth, die durch ihre Rückmeldungen wesentlich zur Verständlichkeit der Arbeit beigetragen haben.

Weiterhin danke ich meinen Kolleginnen und Kollegen für die zahlreichen und interessanten Diskussionen. Mein Dank gilt insbesondere: Marcel Busse, Ursula Eckle, Dirk Farin, Holger Füßler, Thomas Hänselmann, Volker Hilt, Thomas King, Stephan Kopf, Christoph Kuhmünch, Gerald Kühne, Martin Mauve, Michael Möske, Walter Müller, Nicolai Scheele, Claudia Schremmer, Matthias Transier, Jürgen Vogel, Betty Weyerer und Jörg Widmer.

Ebenso bedanke ich mich bei meinen Studentinnen und Studenten.

Schließlich gebührt mein Dank meiner Familie, die in Höhen und Tiefen jederzeit fest zu mir stand und ohne die diese Arbeit niemals entstanden wäre. Danke!

Christian Liebig, im März 2005

Abstract

Distance learners in traditional online exercises and tutoring systems often get stuck with questions for which they need the help of a tutor or colleague. Learning alone can be frustrating. On the other hand we know that real group work is an important component of traditional teaching at all levels of the educational system: It is motivating, the students become active and involved, they enjoy meeting with other students and doing work together. Based on the principle of social constructivism, role-based learning is an especially powerful paradigm of group work.

In a networked virtual environment, the organization of such group work is much more difficult than in face-to-face situations. The organization is very complex and time-consuming. The Communication And Tutoring System (CATS), which was developed in this work, supports the automatic formation of virtual learning groups based on proficiency-levels, as well as the seamless integration of audio/videoconferencing into an tutoring system.

To find the appropriate partner, a measurement algorithm keeps track of the performance level of a learner by measuring the percentage of correct answers at the current level, the reliability with which the learner answers the questions and the time he/she takes. From these measures the system derives a unified performance parameter that controls the presentation and the communication between the students.

Finally a detailed example of group work is given, borrowed from the law school curriculum: “moot court”, a simulated mock court hearing where the students play the roles of prosecutor, defender and judge over the network. The approach is motivated from a pedagogical point of view and realized by a highly integrated technology with videoconferencing, instant messaging and interactive group work.

The CATS system has been used in different subjects and projects. An evaluation of the system shows the benefits for learners as well as for teachers.

Zusammenfassung

In der klassischen Lehre spielen Übungen eine entscheidende Rolle für den Lernerfolg der Studierenden. Insbesondere Gruppenübungen ermöglichen es, einen Wissensgegenstand aus unterschiedlichen Perspektiven zu betrachten und durch das Mittel der Externalisierung besser zu verstehen. Dies setzt allerdings eine direkte Kommunikation zwischen den beteiligten Personen voraus.

Im Bereich der Fernlehre sind diese Möglichkeiten aufgrund der räumlichen Trennung und der technischen Gegebenheiten nur sehr eingeschränkt gegeben. Zum einen existieren entsprechende Systeme nur für Teilaspekte, zum anderen fehlen bisher adaptive integrierte Gruppenübungen.

Das in dieser Arbeit vorgestellte Communication and Tutoring System (CATS) integriert zum einen adaptive Aufgaben mit einem standardisierten Kommunikationssystem und ermöglicht dadurch die Vermittlung von Wissen zwischen den Studierenden untereinander und zwischen Studierenden und Lehrenden. Zum anderen wird ein generischer Ansatz zur technischen Abwicklung von Gruppenarbeiten in der Fernlehre vorgestellt. Sowohl die adaptiven individuellen Übungsaufgaben wie auch das Gruppenarbeitskonzept sind in verschiedenen Fächern einsetzbar. Entsprechende Hilfsprogramme unterstützen die Lehrenden bei der Erstellung von Übungsaufgaben, besondere Programmierkenntnisse sind hierzu nicht erforderlich.

Damit wurde Forderungen aus der Lernpsychologie Rechnung getragen, die bisher, insbesondere im Bereich der Fernlehre, nicht erfüllt werden konnten.

Bereits bei der Architektur des Systems wurde darauf geachtet, eine stabile, nachhaltige Umsetzung des Systems zu gewährleisten, ohne auf eine notwendige Flexibilität zu verzichten.

Ein spezielles Integrationsmodell ermöglicht die rasche Anbindung an bestehende Lernplattformen. Im Rahmen dieser Arbeit wurde die Anbindung an die Lernplattform „LRN“ realisiert.

Das Communication and Tutoring System (CATS) wurde im aktiven Übungsbetrieb zur Unterstützung der Vorlesung Rechnernetze an der Universität Mannheim in den Sommersemestern 2003 und 2004 und auch für die Vorlesung Multimediatechnik im Wintersemester 2003/04 eingesetzt. Es werden die Ergebnisse einer empirischen Untersuchung präsentiert, die zum WS 2003/04 im Rahmen der Vorlesung Multimediatechnik durchgeführt wurde. Diese betraf sowohl die Akzeptanz des Systems bei den Studierenden wie auch die Klausurergebnisse der CATS-Benutzer. Zudem werden die Examensergebnisse von Präsenzstudierenden der Rechnernetze-Vorlesung im SS 2004 mit denen der Fernstudierenden verglichen.

CATS wurde außerdem in den Projekten ULI, VIROR, Winfoline und Politikon erfolgreich eingesetzt.

Inhaltsverzeichnis

Abbildungsverzeichnis	xv
Tabellenverzeichnis	xvii
Abkürzungsverzeichnis	xix
1 Einleitung	1
1.1 Motivation	1
1.2 Problemstellung	2
1.3 Gliederung der Arbeit	2
2 Lernen durch Üben	5
2.1 Grundlagen des Wissens und Lernens	5
2.1.1 Rationalismus	5
2.1.2 Empirismus	6
2.1.3 Epistemologie	6
2.1.4 Synthese der Philosophien	7
2.1.5 Der kartesianische Dualismus	7
2.1.6 Heideggers Theorie	8
2.1.7 Fernöstliche Philosophien	8
2.1.8 Zusammenfassung	8
2.2 Neuzeitliche Definition von Wissen und Lernen	9
2.2.1 Definition von Wissen	9
2.2.2 Definition von Lernen	10
2.3 Lerntheorien	11
2.3.1 Behaviorismus	11
2.3.2 Kognitivismus	13

2.3.3	Konstruktivismus	13
2.4	Modelle zur Wissensvermittlung	14
2.4.1	Der Learning Cycle nach Mayes	14
2.4.2	Vermittlung von Wissensinhalten durch das dynamische Modell nach Nonaka und Takeuchi	16
2.5	Bedeutung von Übungen in der Lehre	20
2.5.1	Pädagogische Motivation der Gruppenarbeit	22
2.5.2	Ausgestaltung von Übungen in der Präsenzlehre	24
2.5.3	Übungen in der Fernlehre	25
2.5.4	Lernziele im universitären Umfeld	28
3	Existierende Vorarbeiten	31
3.1	Animationen und Simulationen	31
3.1.1	Animationen	31
3.1.2	Simulationen	33
3.2	Automatisierte, isolierte Aufgabensysteme	33
3.2.1	Einfache Aufgabensysteme	33
3.2.2	Das Computerunterstützte Adaptive Testen	34
3.3	Intelligente Tutorielle Systeme	35
3.3.1	Aufbau eines ITS	35
3.3.2	Ablauf einer ITS-Sitzung	36
3.3.3	Beispielanwendungen	36
3.3.4	Ein Autorensystem für ITS: Eon	37
3.3.5	Kritik an ITS	38
3.4	Kognitive Tutorielle-Systeme	38
3.5	Systeme zur Unterstützung der sozialen Interaktivität	39
3.5.1	Talking Heads	39
3.5.2	Systeme zur Beurteilung der Lernleistung von Studierenden	39
3.5.3	JaTeK als Lernplattform für interaktives Lernen	42
3.6	Systeme zur Unterstützung von Lerngruppen	43
3.6.1	Praktomat	43
3.6.2	VITAL	43
3.6.3	CROCODILE	44
3.6.4	ExpertFinder	44
3.7	Autorensysteme zur Erstellung von Übungsaufgaben	44

3.7.1	Cognitive Tutor Authoring Tools	45
3.7.2	Der WIL-MA-Editor	45
3.7.3	Unterstützung von Übungsaufgaben durch IMS QTIS	45
3.7.4	Zusammenfassung	46
3.8	Integrierte Lern- und Übungsumgebungen	47
3.8.1	SCORM-Die Austauschnorm für Lerninhalte	47
3.8.2	CLIX-Campus	50
3.8.3	WebCT	51
3.8.4	Das Blackboard Learning System	51
3.8.5	Lernraum virtuelle Universität	52
3.8.6	Die Lernplattform .LRN	53
3.9	Systeme zur Unterstützung der synchronen Lerner-Kommunikation .	55
3.9.1	Audio-/Video-Konferenz-Systeme	56
3.9.2	Instant-Messenger-Dienste	56
3.10	Zusammenfassung	58
4	Anforderungen an ein Kommunikations- und Tutoring System für Lern-	
	gruppen	59
4.1	Unabhängigkeit von Wissensgebieten	59
4.2	Adaptivität an den Leistungsstand der Studierenden	59
4.3	Kommunikationsunterstützung	60
4.4	Unterstützung von Gruppenarbeit	60
4.5	Allgemeine Anforderungen	61
4.6	Didaktische Anforderungen	62
4.7	Wirtschaftliche Anforderungen	62
4.8	Bewertbarkeit der Lehr- und Lernleistung	62
4.9	Allgemeine Verbesserungsmöglichkeiten	65
4.9.1	Verbesserungsansätze des Beurteilungssystems	65
4.9.2	Organisatorische Anforderungen	66
5	Architektur des Communication and Tutoring System (CATS)	69
5.1	Grundkonzeption	69
5.2	Individuelle, adaptive Aufgaben: Szenario 1	70
5.2.1	Kennzahlensystem zur Wissensbewertung	70
5.2.2	Adaptivität	73
5.2.3	Realisiertes Szenario 1: Adaptive, individuelle Übungsaufgaben	74

5.3	Die Peer-to-Peer Kommunikationsanbindung in CATS: Szenario 2 . . .	74
5.3.1	H.323-konforme Kommunikation in CATS	75
5.3.2	Das Session Initiation Protocol SIP	77
5.3.3	Bewertung und Testat der individuellen Leistungsbeiträge . . .	80
5.3.4	Realisiertes Szenario 2: Unterstützung der Kommunikation zwischen Lernern	81
5.4	Unterstützung von Lernergruppen durch CATS: Szenario 3	82
5.4.1	Das CATS-Auswahlverfahren zur Gruppenbildung	83
5.4.2	Die Erstellung einer adaptiven Gruppenaufgabe	85
5.4.3	Die Gruppenarbeitsarchitektur	86
5.4.4	Die Gruppenkommunikation	88
5.4.5	Möglichkeiten der Gruppenbewertung	91
5.4.6	Realisiertes Szenario 3: Unterstützung von Lernergruppen . . .	92
5.5	Die CATS-Aufgaben-Autorenwerkzeuge	93
5.5.1	Der Multiple-Choice-Editor	93
5.5.2	Der CATS-Aufgabeneditor für projektbezogene Aufgaben . . .	94
5.6	Die Client-Server-Architektur	98
5.7	Der CATS-Server	99
5.7.1	Die Datenbankschicht von CATS	100
5.7.2	Die Anwendungsschicht	101
5.8	Der CATS-Client	101
5.9	Integrationsmöglichkeiten von CATS in bestehende Lernplattformen .	102
5.9.1	Das Integrationsmodell von CATS	103
5.9.2	Die Integrationsmöglichkeiten	104
6	Die Implementierung des Communication and Tutoring Systems (CATS)	107
6.1	Die Implementierung des CATS-Servers	107
6.1.1	Das CATS-Datenbanksystem	107
6.1.2	Die CATS-Anwendungsschicht	110
6.2	Die Implementierung der CATS-Clients	113
6.2.1	Grundlagen des Usability-Design	113
6.2.2	Design des Benutzerinterfaces	117
6.2.3	Die CATS-Aufgabenmodule	117
6.3	Das CATS-Autorenwerkzeug für projektbezogene Aufgaben	121
6.4	Die Kommunikationsunterstützung	121

6.4.1	Die Peer-to-Peer-Kommunikation	122
6.4.2	Unterstützung der Gruppenkommunikation durch eine MCU .	124
6.4.3	SIP-Integration	124
6.5	Die CATS-Administration	125
6.5.1	Die Systemadministration	125
6.5.2	Der Benutzerbereich	126
6.6	Anbindung von CATS an Lernplattformen	127
6.6.1	Anwendung des CATS-Integrationsmodells	127
6.6.2	Anbindung von CATS an .LRN	128
6.7	Zusammenfassung	128
7	Die Evaluierung des CATS-Systems	131
7.1	Anwendungen aus dem Bereich der Informatik	132
7.1.1	CATS-Applets zum Thema Rechnernetze	132
7.1.2	CATS-Applets zu Themen der Multimediatechnik	141
7.2	Beispiele aus dem Bereich der empirischen Sozialforschung	151
7.2.1	Übungsapplet zur Randomized Response Technik (RRT)	151
7.2.2	CATS-Applet zum Non-Equivalent-Group-Design	152
7.3	Die simulierte Gerichtsverhandlung	153
7.4	Evaluierung in verschiedenen Lehr-/Lernszenarien	155
7.4.1	Rahmenbedingungen des Experiments	156
7.4.2	Hypothesen	156
7.4.3	Experiment-Design	157
7.4.4	Ergebnisse des Experimentes	157
7.4.5	Examensergebnisse im Fach Rechnernetze aus dem SS 2004 . .	160
7.4.6	Diskussion der Ergebnisse der empirischen Evaluierung	161
7.5	Zusammenfassendes Ergebnis der Evaluierung	161
8	Zusammenfassung und Ausblick	163
8.1	Zusammenfassung	163
8.2	Ausblick	164
	Literaturverzeichnis	167
	Index	183

Abbildungsverzeichnis

2.1	Übersicht über die Erkenntnisphilosophien	9
2.2	Der Learning Cycle nach [MCTR94]	15
2.3	Modell zur Wissensumwandlung nach [NT97][S. 75]	17
2.4	Modell zur Wissensumwandlung nach [NT97][S. 84]	20
2.5	Spirale der Wissensschaffung in der Universität nach [NT97][S. 87] . .	21
2.6	C_SCL nach Raum/Zeit nach [Mit98][S. 22]	26
3.1	Einordnung von Aufgabeneditoren	46
3.2	Die SCORM-Kurs-Architektur	48
3.3	Die .LRN-Lernplattform an der Universität Mannheim	54
5.1	Offline-Szenario von CATS	74
5.2	Der Aufbau von H.323 nach [Aro00]	75
5.3	Die SIP-/CATS-Integration nach [Bei03][S.29ff.]	80
5.4	Online-Übung mit zweifacher Rückkopplung	82
5.5	Informationsübersicht Wissen pro Lernziel pro Studierenden	83
5.6	Bildung von Teams	84
5.7	Aufteilung der Lerngemeinschaft in Arbeitsgruppen und Rollen bzw. Untergruppen	85
5.8	Berechnung des Gruppenschwierigkeitsgrades	85
5.9	Das CATS-Gruppenarbeitsobjekt	87
5.10	Die Organisation der Gruppenrechte am Beispiel des Moot-Courts . .	88
5.11	Die mehrdimensionale Gruppenkommunikation	90
5.12	Gruppenübung mit Kommunikationsunterstützung	92
5.13	Der WIL-MA-Multiple Choice Editor	94
5.14	Das Multiple-Choice Applet	95

Abbildungsverzeichnis

5.15	Modifizierte Netzplantechnik in Objektform	96
5.16	Die Oberfläche des Aufgabeneditors	97
5.17	Der Backtrainer	98
5.18	Die Systemstruktur von CATS	99
5.19	Die Benutzerschnittstelle von CATS	102
5.20	Das CATS-Integrationsmodell	103
6.1	Das Entity-Relationship-Diagramm der CATS-Datenbank	109
6.2	Die CATS-Anwendungen	111
6.3	Das CATS-Verwaltungssystem	113
6.4	Statistische Auswertungen durch das EACS	114
6.5	Die CATS-Navigationsleiste	115
6.6	Die Integration des Result-Objektes in eine Übungsaufgabe	119
6.7	Der Arbeitsbereich der Rollen im Moot-Court	120
6.8	Die Erstellung einer dynamischen projektbezogenen CATS-Aufgabe	122
6.9	Die CATS-Awarenessfunktion zur Peer-Kommunikation	123
6.10	Die CATS-Studierenden Registration	127
6.11	Die CATS/.LRN-Integration: Ergebnis-Anzeige	129
7.1	Das Line-Coding Applet	133
7.2	Das CATS-Applet zum Thema "Token Ring"	135
7.3	Probleme im Medienzugriff bei WLAN	135
7.4	Das WLAN-Übungsapplet	136
7.5	Das CATS-Applet zum Routing-Verfahren nach Dijkstra's Algorithmus für Kürzeste Pfade	138
7.6	Das Multicast-Simulations- und Übungsapplet	139
7.7	Das Congestion-Control-Applet: Berechnungsaufgabe	140
7.8	Das Congestion-Control-Applet: Analyseaufgabe	141
7.9	Das Congestion-Control-Applet: Zeichnung	142
7.10	Der Kognitive Tutor zur Huffman-Codierung	145
7.11	Die Übungsumgebung zur Fouriertransformation	146
7.12	Die Übungsumgebung zur diskreten Kosinustransformation	147
7.13	Navigation über die einzelnen Verfahren	149
7.14	Navigation im Ablauf des Verfahrens	150
7.15	CATS-Applet zur Übung der Randomized Response Technik	152
7.16	Das Non-Equivalent-Group-Design-Applet	153

Tabellenverzeichnis

5.1	Bewertung der Wichtigkeit einzelner KPIs	72
6.1	Variablen der Result-Klasse	118
7.1	Ergebnisse des Akzeptanztests	158
7.2	Klausurergebnisse nach CATS-Gruppen im WS 2003/04	160

Abkürzungsverzeichnis

Symbole

.LRN Eigenname der .LRN-Lernplattform

A

ACT Adaptive Control of Thought

ADL Advanced Distributed Learning Initiative

ADSL Asymmetric Digital Subscriber Line

AOL America Online

Apache Eigenname eines Web-Servers und Akronym für "A patchy server"

API Application Programming Interface

ARIADNE Alliance of Remote Instructional Authoring and Distribution Networks

B

BWL Betriebswirtschaftslehre

BYE SIP-Nachricht: Beendigungswunsch

C

C	Eigenname der Programmiersprache C
C++	Eigenname der Programmiersprache C++
C-Scan	Circular SCAN
CACSD	Computer-Aided Control Systems Design
CANCEL	SIP-Nachricht: Abbruch einer bestehenden Verbindung
CAT	Computerunterstütztes Adaptive Testen
CATS	Communication and Tutoring System
CBAT	Content Balanced Adaptive Testing
CBT	Computer Based Training
CD	Compact Disc
CLIX	Corporate Learning & Information eXchange
Conn	Connection
cos	Cosinus
CPL	Call Processing Language
CPM	Critical Path Method
CROCODILE	Creative Open Cooperative Distributed Learning Environment
CSCL	Computer Supported Cooperative Learning
CSCMA/CA	Carrier Sensing Multiple Access with Collision Avoidance
CSCW	Computer Supported Cooperative Work
CTAT	Cognitive Tutor Authoring Tools
CTS	Clear To Send

cwnd congestion window

D

DBMS Database Management System

DCF Distributed Coordination Function

DCT Discrete Cosine Transform

DFWMAC Distributed Foundation Wireless Medium Access Control

DIFS Distributed Coordination Function Inter-Frame Spacing

DVD Digital Versatile Disc

E

E-Mail Electronic Mail

EACS Exercise Administration and Controlling System

EDV Elektronische Datenverarbeitung

Eon Ein Autorensystem zur Erstellung von Lernumgebungen

EXCHECK Eigenname eines ITS

ExpertFinder Eigenname eines Systems zum Auffinden von Mitlernern

F

F-15 Eigenname eines Kampflugzeugs

FCFS First Come First Serve

FH Fachhochschule

G

G.700	Norm zur Übertragung von Audio-Signalen
G.728	Norm zur Übertragung von Audio-Signalen
GLC	Global Learning Consortium
GmbH	Gesellschaft mit beschränkter Haftung
GNU	Rekursives Akronym für "GNU is Not UNIX"
GSPT	General Systems Performance Theory

H

H.225	Norm zur Verhandlung von Übertragungsparametern
H.261	Norm zur Übertragung von Video-Signalen
H.263	Norm zur Übertragung von Video-Signalen
H.264	Norm zur Übertragung von Video-Signalen
H.323	Übertragungsnorm zur Videokonferenz
HRG	Hochschulrahmengesetz

I

ICQ	Akronym für I seek you
ID	Identifizier
IEEE	Institute for Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IMS	Instructional Management Standards
INVITE	SIP-Nachricht: Sitzungswunsch

IP	Internet Protocol
IRT	Item Response Theory
ISDN	Integrated Service Digital Network: Digitales, integriertes Kommunikationssystem
ISO	International Standard Organisation
ISP	Internet Service Provider
ITS	Intelligente Tutorielle Systeme
ITU-T	International Telecommunication Union-Telecommunication-Standardization Bureau

J

JaTeK	Java Based Teleteaching Kit
JaWoS	Java Based Workgroup Support
JDBC	Java Database Connectivity: Markenzeichen der Firma Sun
JETS	Java Enabled Telecollaboration System

K

kBit	Kilo bit
KI	Künstliche Intelligenz
KPI	Key Performance Indicators

L

LOM	Learning Object Metadaten Model
LVU	Lernraum Virtuelle Universität

M

MBone	Multicast Backbone
MCU	Multipoint Control Unit
MIT	Massachusetts Institute of Technology
MoRob	Modular Educational Robotic Toolbox
MPL	Mozilla Public Licence
MPM	Metra Potential Method
MS	Eigenname der Firma Microsoft
MSI	Mathematical Science Inventory
MSN	Microsoft Network

N

NPT	Netzplantechnik
------------	-----------------

O

ODBC	Open Database Connectivity
OpenACS 5	Eigenname: Open Architecture Community System
OSI	Open Sytem Interoperatibility

P

PC	Personal Computer
PCF	Point Coordination Function
PDU	Packet Data Unit
PERT	Program Evaluation and Review Technique

PHP	Rekursives Akronym für 'PHP: Hypertext Preprocessor'
PIFS	Point Coordination Function Inter-Frame Spacing
PIM	Protocol Independent Multicast
PolitikON	Politikwissenschaft online
PostgreSQL	Eigenname eines Database Management Systems

Q

Q.931	Norm zur Regelung des Verbindungsaufbaus
QTIS	Question & Test Interoperability Specification

R

RAID	Reuseability Accessibility Interoperability Durability
REGISTER	SIP-Nachricht: Registrierung beim Server
RFC	Request for Common
RRT	Randomized Response Technik
RTCP	Realtime Protocol Control Protocol
RTP	Realtime Protocol
RTS	Request To Send

S

s	second
S-R	Stimuli Reactio
SCO	Sharable Content Object
SCORM	Sharable Content Object Reference Model

Tabellenverzeichnis

SDK	Software Development Kit
SHERLOCK	Eigenname eine ITS
SIFS	Short Inter-Frame Spacing
SIP	Session Initiation Protocol
SK8	Eigenname einer Programmiersprache
SMS	Short Message Service
SQL	Structured Query Language
SS	Sommersemester
SSTF	Shortest Seek Time First
ssthresh	Slow-Start Threshold
SuSE	Markenname der SUSE Linux AG
SWS	Semesterwochenstunden

T

T.120	Norm zur Fernsteuerung von Applikationen
TAO	Tactical Actions Officer
TCP	Transmission Control Protocol
TDACK	Triple Duplicate Acknowledgement

U

UDP	User Datagram Protocol
ULI	Universitärer Lehrverbund Informatik
URI	Unified Ressource Identifier

URL Unified Ressource Locator

V

VALID Eigenname eines ITS

VB Visual Basic

VIROR Virtuelle Hochschule Oberrhein

VITAL Virtual Teaching and Learning

VoIP Voice over Internet Protocol

VU Virtuelle Universität

W

WebAssign Eigenname, Name einer Übungsplattform

WebCT Eigenname: Web Course Tools

WebDAV Eigenname: WWW Distributed Authoring and Versioning

WIL-MA Wireless Interactive Learning -Mannheim

Winfoline Wirtschaftsinformatik online

WLAN Wireless Land

WS Wintersemester

WWW World Wide Web

X

XLX eXtreme e-Learning eXperience: Eigenname einer Lernplattform der Universität Münster

XML Extensible Markup Language

1 Einleitung

Die Ressource Wissen entwickelt sich zu einem strategischen Gut. Wissen gilt inzwischen als maßgebender Produktions- und Wirtschaftsfaktor (siehe hierzu [Krc00], [Bul99]). Daher gewinnt insbesondere auch die Wissensvermittlung durch Lernvorgänge an Bedeutung. Es ist jedoch zu berücksichtigen, dass Wissen einerseits räumlich verteilt und zu unterschiedlichen Zeiten generiert und andererseits an entfernten Orten zu wiederum anderen Zeiten benötigt wird. Diese Umstände wurden insbesondere in den letzten Jahren durch die Globalisierung verstärkt.

Die klassische, synchrone Wissensvermittlung, wie sie an zentralen Orten (z. B. Universitäten, Weiterbildungseinrichtungen etc.) stattfindet, ist mit den gebräuchlichen Methoden des lokalen Frontalunterrichtes nicht in der Lage, diese Aufgabe effizient zu erfüllen. Es ist demnach eine Herausforderung, dieses Wissen auf effiziente Weise zu verteilen und hierbei räumliche und zeitliche Barrieren zu überwinden.

Die Universitäten haben hierbei eine Schlüsselrolle. Sie sind in zweifacher Richtung gefordert: Zum einen als Erzeuger von Wissen im Bereich der Forschung und gleichzeitig als Vermittler von Wissen im Bereich der Lehre.

1.1 Motivation

Diese Arbeit entstand im Rahmen des Projektes “Universitärer Lehrverbund Informatik” (ULI) [KK], [FG04][S. 310f.] In diesem Projekt wurden am Lehrstuhl für Praktische Informatik IV der Universität Mannheim zwei Vorlesungen (“Multimedia-technik” und “Rechnernetze”) zu Fernlehrvorlesungen weiterentwickelt und die Betreuung der Fernstudierenden an insgesamt elf in Frage kommenden Standorten sichergestellt. Es zeigte sich zu Beginn des Projektes, dass zwar die Inhaltsaufbereitung, z. B. durch Vorlesungsaufzeichnungen, bereits gut verstanden war und eine breite Produktpalette zur Verfügung stand, dies traf jedoch nicht für den Bereich der Übungsbetreuung zu. Hier existiert zwar eine Vielzahl an Übungsplattformen zur Unterstützung von Multiple-Choice-Aufgaben oder zur organisatorischen Übungs-

1 Einleitung

abwicklung, aber diese Systeme erwiesen sich als zu statisch und keinesfalls zur Unterstützung von netzbasierten Gruppenarbeiten, insbesondere im Hinblick auf eine Kommunikationsunterstützung, geeignet. Dies war auch der Grund dafür, dass der Forschungsschwerpunkt im Mannheimer Beitrag zum Projekt ULI in der Entwicklung von neuen Technologien lag, die das Lernen in Gruppen unterstützt und hierbei insbesondere neue Formen der Kommunikation anbietet.

1.2 Problemstellung

Es war daher ein System zu entwickeln, aufbauend auf Erkenntnissen der pädagogischen Psychologie, welches in der Lage ist, Übungen in der Fernlehre abzubilden. Hierbei sollte das System die synchrone Kommunikation der handelnden Personen (Studierende untereinander wie auch Studierende zu Dozierenden) untereinander unterstützen und insbesondere die Kommunikationslücke, die bedingt durch die Asynchronität der Fernlehre entsteht, reduzieren. In einem weiteren Schritt sollte eine Architektur erarbeitet werden, welche in der Lage ist, Gruppenarbeiten im asynchronen Lehrmodus abzubilden. Hierbei sollte insbesondere die Gruppenarbeit als eine solche gemeinsame Arbeit verstanden werden, in der die einzelnen Gruppenmitglieder Rollen einnehmen und gemeinsam an einer Problemlösung arbeiten können.

1.3 Gliederung der Arbeit

Die Arbeit gliedert sich wie folgt: Ausgehend von theoretischen Grundlagen aus dem Bereich der Erkenntnistheorie werden die modernen Lerntheorien dargelegt. Es erfolgt eine detaillierte Problembeschreibung im Bereich der universitären Wissensvermittlung in Übungen, hierbei insbesondere im Fernlehrmodus. Es werden existierende Lösungsansätze beschrieben und darauf aufbauend die System-Anforderungen an das zu erstellende System formuliert. Anschließend wird die Architektur des entwickelten "Communication and Tutoring System" (CATS) vorgestellt. Danach folgt ein Kapitel über die Implementierung, in der die technische Realisierung detailliert beschrieben wird.

Im folgenden Kapitel Evaluierung werden Anwendungsbeispiele aus unterschiedlichen Wissensgebieten demonstriert und die Ergebnisse einer empirischen Untersu-

chung des CATS-Systems im Wissensgebiet "Multimediatechnik", welche im Wintersemester 2003/04 durchgeführt wurde, vorgestellt. Desweiteren werden die Examensergebnisse der Vorlesung Rechnernetze im SS 2004 mit Hinblick auf die CATS-Nutzung diskutiert.

Die Arbeit schließt mit einer Zusammenfassung sowie einem Ausblick auf mögliche, zukünftige Erweiterungen.

2 Lernen durch Üben

Dieses Kapitel erörtert die theoretischen Grundlagen des Lernens, die wichtigsten Lerntheorien und geht insbesondere auf die Bedeutung von Übungen in der klassischen universitären Ausbildung ein. Es werden zunächst die relevanten Erkenntnistheorien, ausgehend vom Rationalismus bis hin zu fernöstlichen Theorien vorgestellt. Hierauf aufbauend werden die neuzeitlichen Definitionen von Wissen und Lernen erläutert und anschließend die Umsetzung durch verschiedene Lerntheorien beschrieben. Hier haben sich verschiedene Modelle der Wissensvermittlung, wie z. B. der Learning-Cycle nach Mayes et al. oder die Wissensspirale von Nonaka und Takeuchi herausgebildet. Das Kapitel schließt mit der Beschreibung von Übungen in der Präsenzlehre und erörtert die Problembereiche von Übungen im Fernlehrszenario.

2.1 Grundlagen des Wissens und Lernens

Die bekannten Lerntheorien haben ihre Wurzeln in unterschiedlichen theoretischen Grundlagen. Innerhalb der Philosophie wurden insbesondere die Begriffe "Erkenntnis", "Wissen" und "Lernen" diskutiert. Daher ist es im Rahmen einer Arbeit, die ein System zur Unterstützung von Lerngruppen zum Gegenstand hat, notwendig, diese Grundlagen zu erklären. Hierbei stehen jedoch die westlichen Philosophien im Mittelpunkt.

2.1.1 Rationalismus

Der Rationalismus geht davon aus, dass sich Wissen durch logisches Denken deduktiv erschließen lässt. Hierbei existiert apriorisches Wissen, welches nicht durch Sinneseindrücke erklärt werden muss. Die Wahrheit wird durch logisches Denken erschlossen, das wiederum auf Axiomen beruht [NT97][S. 33].

Kontinentaler Rationalismus

Descartes begründete durch sein Buch “Discours de la méthode” von 1637 [Des37] den kontinentalen Rationalismus und stellte folgende Regeln des rationalen Denkens auf [Rus97][S. 275]:

1. Niemals eine Sache als wahr anzunehmen, die ich nicht als solche sicher und einleuchtend erkennen würde, das heißt sorgfältig die Übereilung und das Vorurteil zu vermeiden und in meinen Urteilen nur so viel zu begreifen, wie sich meinem Geist so klar und deutlich darstellen wird, dass ich gar keine Möglichkeit hätte, daran zu zweifeln.
2. Jede der Schwierigkeiten, die ich untersuchen würde, in so viele Teile zu zerlegen als möglich und zur besseren Lösung wünschenswert ist.
3. Meine Gedanken zu ordnen; zu beginnen mit den einfachsten und fasslichsten Objekten und aufzusteigen allmählich und gleichsam stufenweise.

Weitere Vertreter dieser Philosophie waren neben Descartes unter anderem Leibniz und Pascal [Höf01][S. 149ff.].

2.1.2 Empirismus

Diese Philosophie des Rationalismus wurde jedoch von John Locke, dem Begründer des britischen Empirismus, kritisiert. Er postuliert in seinem Werk “An Essay Concerning Human Understanding”[Loc90] die Objektivität der Dinge, die in der Welt existieren [NT97][S. 36]. Locke beschreibt den menschlichen Geist als eine “tabula rasa” und widerspricht damit dem Rationalismus, welcher konstatiert, dass wahres Wissen aus einem apriorischen Wissen durch logisches Schließen erlangt werden kann. Nach Locke jedoch kann nur die Erfahrung den Geist mit Ideen versorgen. Hierbei existieren zwei Arten von Erfahrung: Die Sinneswahrnehmung und die Reflexion. Die Sinneswahrnehmungen werden als einzige Quelle des Wissens betrachtet.

2.1.3 Epistemologie

Vertreter der Epistemologie sind unter anderem Plato und Aristoteles. In dem so genannten Höhlengleichnis erscheint Plato die materielle Welt nur als Schatten einer

vollkommenen Welt der Ideen. Die Menschen streben nach diesen ewigen, unveränderlichen Ideen, die nicht durch Sinneswahrnehmungen, sondern nur durch die reine Vernunft erkannt werden. Gegenüber Plato betont Aristoteles hingegen die Bedeutung der Beobachtung und die dadurch mögliche Verifikation der einzelnen Sinneswahrnehmungen[NT97] [S. 32ff.].

2.1.4 Synthese der Philosophien

In einem weiteren Schritt versuchen Kant, Hegel und Marx, den Rationalismus und den Empirismus zusammenzuführen. Hierbei sieht Kant den menschlichen Verstand nicht als passive "tabula rasa", sondern als aktive Instanz, die die Sinneseindrücke in Zeit und Raum ordnet und zu ihrem Verständnis Begriffe und Instrumente bereithält. Kant glaubt jedoch, dass wir nicht das "Ding an sich" kennen können, sondern nur seine Wahrnehmung das "Phänomen". Die Philosophie wird als transzendentaler Idealismus bezeichnet. Hegel vertritt die Ansicht, dass sowohl Verstand wie auch Materie durch einen dialektischen Prozess vom "absoluten Geist" abgeleitet sind.

Die Definition von Dialektik nach Hegel lautet: "Dialektik ist die Synthese, die These und Antithese miteinander versöhnt und sich in diesem Prozess von allen nichtrationalen Elementen löst." [NT97][S. 24ff.].

Marx kombinierte schließlich Hegels Dialektik und die Sozialwissenschaft in einer eigenen Theorie. Hierbei tritt jedoch ein Widerspruch zu Hegels idealistischer Philosophie auf: Die Begründung lag in der Unmöglichkeit der Klärung der dynamischen Beziehung zwischen Mensch und Umwelt. Nach Marx erfolgt durch das Streben nach Erkenntnis eine dialektische Wechselwirkung zwischen Subjekt und Objekt.

2.1.5 Der kartesianische Dualismus

Dem kartesischen Dualismus liegt die Annahme zugrunde, dass das Wesen des Menschen in seinem rationalen Denken liegt. Bedingt durch die Abspaltung des denkenden Ichs von seiner Umwelt strebt es nach Wissen. Die zeitgenössische Auseinandersetzung betont, dass Erkenntnis eine Wechselwirkung zwischen dem Ich und der Außenwelt voraussetzt [NT95][S. 25].

2.1.6 Heideggers Theorie

Demgegenüber geht Heidegger davon aus, dass das "In-der-Welt-Sein" dem Menschen kein losgelöstes denkendes Ich erlaubt, sondern nur eine enge Beziehung zwischen Wissen und Handeln. Dies steht im Widerspruch zum kartesischen Dualismus. Eine Betonung zwischen Wissen und Handeln findet dann durch den Existenzialismus statt. Die meisten Phänomenologen gehen davon aus, dass Wissen durch Reflexion erlangt werden kann. Demgegenüber behaupten die Existenzialisten, dass man die Welt nur durch zweckbezogenes Handeln erkennen kann [NT97][S.38f.]. Dies findet zum Beispiel auch seinen Niederschlag in der Lerntheorie des Konstruktivismus, der in Abschnitt 2.3.3 näher behandelt wird.

2.1.7 Fernöstliche Philosophien

Es sollte nicht unerwähnt bleiben, dass "Wissen" auch aus einer anderen philosophischen Richtung betrachtet werden kann. Hierbei spielen die fernöstlichen Philosophien eine wichtige Rolle. Die japanische Tradition zum Beispiel geht von einem Einheitsgedanken von Mensch und Natur aus. Es besteht zudem eine andere Auffassung über Zeit und Raum. Die japanische Zeitauffassung ist zirkulär und auf den Augenblick bezogen. Die Japaner begreifen die Zeit als einen kontinuierlichen Fluss einer immer wieder neuen Gegenwart [NT97][S. 41] gegenüber dem sequentiellen, westlichen Zeitbild. Auch sind Unterschiede in der Raumauffassung festzustellen. Dies wird zum Beispiel in bildlichen Darstellungen sichtbar: Bilder verfügen nicht über eine Perspektive, die den Standort ermittelbar macht. Japaner berufen sich tendenziell weniger auf abstrakte oder metaphysische Theorien, sondern mehr auf ihre eigene Erfahrungswelt.

2.1.8 Zusammenfassung

Es bleibt festzuhalten, dass sich seit der Antike die Philosophie mit der Definition von "Erkenntnis" und "Wissen" beschäftigt hat. Abbildung 2.1 gibt hierzu einen abgeleiteten Überblick. Im Laufe der Zeit wurden die beiden grundlegend verschiedenen Theorien des Rationalismus und Empirismus vereint. Schließlich entwickelte sich die Philosophie, dass sich Wissen zwar einerseits durch Erfahrung und durch logisches Schließen ergibt, aber dass zudem noch ein sozialer Austausch erforderlich ist.

2.2 Neuzeitliche Definition von Wissen und Lernen

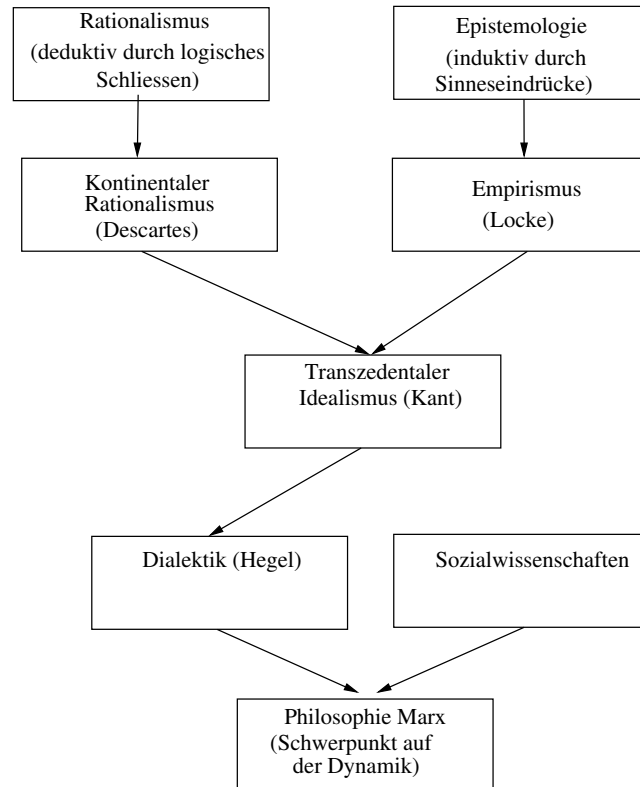


Abbildung 2.1: Übersicht über die Erkenntnisphilosophien

Wie in Abschnitt 2.3 noch zu sehen sein wird, beruhen zum Teil die Lerntheorien auch auf diesen philosophischen Erkenntnissen.

2.2 Neuzeitliche Definition von Wissen und Lernen

Basierend auf den philosophischen Grundlagen haben sich verschiedene Definitionen von "Wissen" und "Lernen" herausgebildet.

2.2.1 Definition von Wissen

"Knowledge can mean information, awareness, knowing, cognition, sapience, cognizance, science, experience, skill, insight, competence, know-how, practical ability, capability, learning, wisdom, certainty, and so on. The definition depends on the context in which the term is used" [Sve97][S. 26]

Wie dieses Zitat zeigt, gibt es vielfältige Interpretationsmöglichkeiten für das, was

2 Lernen durch Üben

man als "Wissen" bezeichnet. Es ist der jeweilige Anwendungskontext zur Interpretation hinzuzuziehen, vgl. hierzu [Wei02b][S. 24f.]. Im Rahmen der wissenschaftlichen Ausbildung ist "Wissen" zudem in eine kollektive Wissensgemeinschaft eingebettet (so genannte Community). Demnach muss insbesondere das individuelle Wissen gegenüber dem kollektiven Wissen unterschieden werden. Dies ist für diese Arbeit insofern von Bedeutung, als die Lerninhalte möglichst auf neuestem Stand von Wissenschaft und Technik gehalten werden sollten. Hierbei sind insbesondere die Erkenntnisse, die durch die wissenschaftliche Gemeinschaft im Rahmen von Publikationen und Konferenzen vermittelt werden, fortlaufend zu beachten.

Eine weitere Unterscheidung des Wissens ist die Einordnung in *Wissen erster Ordnung*, das heißt, hierbei wird Wissen sowohl als Objekt wie auch als Prozess verstanden, es ist damit gegenstandsgebunden, und in das *Wissen zweiter Ordnung* (so genanntes "Meta-Wissen"), welches zur Verarbeitung und Steuerung des Wissens erster Ordnung verwendet werden kann. Dies spielt z. B. beim selbstgesteuerten Lernen eine wichtige Rolle oder bei der Suche nach "Wissensträgern". Diese Unterscheidung spielt insbesondere im Rahmen der Architektur des Gesamtsystems eine wesentliche Rolle.

Eine wichtige Unterscheidung, die auch für die Wissensvermittlung von entscheidender Bedeutung ist, ist die Unterscheidung in *implizites* und *explizites* Wissen der einzelnen Person. Implizites Wissen ist persönlich, erfahrungsgebunden und kontextspezifisch. Es betrifft meist innere Werte, Emotionen und Überzeugungen und ist daher nur schwer kodifizierbar und somit nur schwer durch Medien übertragbar; vielmehr kann dieses Wissen selbst nur in einem sozialen Kontext und daher nur durch den Menschen selbst übermittelt werden. So ist es beispielsweise möglich, innere Werte durch die Darstellung von Emotionen und Gefühlen adäquat zur transportieren. Dies ist u. a. auch ein Grund, warum sich die so genannten Emotion-Icons (z. B. "Smiley") im Bereich der E-Mail Kommunikation großer Beliebtheit erfreuen. Das explizite Wissen hingegen ist unabhängig von einzelnen Personen durch eine formale, systematische Sprache transferierbar. Klassische Medien wie z. B. Bücher oder Filme sind geeignet, explizites Wissen zu übermitteln.

2.2.2 Definition von Lernen

Eines der primären Ziele des in dieser Arbeit entwickelten CATS-Systems ist die Unterstützung des Lernens, wobei allerdings der Begriff "Lernen" auch in diesem

Kontext sehr unterschiedlich interpretiert wird.

Im Alltag wird "Lernen" meist als notwendige Voraussetzung zur Leistungssteigerung (wie z. B. in der Schule) gesehen.

In der Psychologie hingegen ist Lernen eine "andauernde Verhaltensänderung, die nicht durch Ermüdung, Reifung oder chemische Einflüsse zustande gekommen ist" [Ang78][S. 26]. Eine neuere Definition des Lernens in Anlehnung an [Zim92][S. 227] lautet: "Lernen ist ein Prozess, der zu relativ stabilen Veränderungen im Verhalten oder im Verhaltenspotential führt und auf Erfahrung aufbaut."

Allgemein wird heute Lernen als "...die Fähigkeit charakterisiert ..., bestehende Denk- und Handlungsmuster zu revidieren und sich dabei zielgerichtet Veränderungen anzupassen bzw. neue Orientierungen zu entwickeln." [Wei02a].

Die letzte Definition soll als Grundlage für diese Arbeit gesehen werden. Den Studierenden soll durch das CATS-System eine entsprechende Unterstützung zum selbst-gesteuerten Lernen im genannten Sinne gegeben werden.

2.3 Lerntheorien

In der Pädagogischen Psychologie haben sich, basierend auf den oben vorgestellten Philosophien, vier wesentliche Lerntheorien herausgebildet:

2.3.1 Behaviorismus

Der Behaviorismus, der von Pawlow [Paw53] und Skinner [Ski74] begründet wurde, geht davon aus, dass sich das Verhalten durch äußere Hinweisreize und Verstärkungen steuern lässt. Hierbei lassen sich S-R-Verbindungen (Stimuli und Reactio) in Ketten aneinander reihen. Die Realisierung erfolgt zumeist durch Präsentation von Information, Abfrage und Rückmeldung. Das Ziel ist der Erwerb von Faktenwissen.

Der Behaviorismus wird noch einmal in zwei weitere Modelle unterteilt: Die Klassische Konditionierung, dessen Hauptvertreter Pawlow ist, und das Modell des operanten Konditionierens, welches von Skinner entwickelt wurde.

Klassische Konditionierung

Das Modell der klassischen Konditionierung lässt sich am bekannten Beispiel der Pawlowschen Hunde demonstrieren: Pawlow ließ vor der Fütterung von Hunden

2 Lernen durch Üben

eine Glocke läuten (unkonditionierter Reiz). Während der Fütterung registrierte er einen erhöhten Speichelausfluss der Hunde (eine unkonditionierte Reaktion auf den Glockenton). Nach einer Trainingsphase stellte sich jedoch der Speichelfluss bereits bei Erklängen des Glockentons, ohne Futtergabe, ein. Die Hunde hatten eine Reiz-Reaktions-Verbindung gelernt, der zuvor unkonditionierte Reiz wurde zu einem konditionierten.

Durch die räumlich-zeitliche Verknüpfung eines unkonditionierten Stimulus (Futtergabe) mit einem weiteren, neutralen Reiz (Glockengeräusch) wird eine neue Reiz-Reaktions-Verbindung gelernt, so dass der neue Reiz (Glockengeräusch) die gleiche Reaktion (Speichelfluss) hervorruft.

Operantes Konditionieren

Während bei Pawlow noch die einfache Verknüpfung von Reizen mit raum-zeitlich nahe liegenden Reaktionen im Mittelpunkt stand, betrachtete Skinner zudem noch die Konsequenzen, die mit dieser Handlung verbunden sind. Es entsteht in diesem Modell eine Rückkopplung des Stimulus auf den Organismus selbst, und dieser löst damit wieder Reaktionen aus. Hierauf wiederum reagiert die Umwelt entweder zustimmend oder ablehnend. In Abhängigkeit dieser Reaktionen wird der Lernende in ähnlichen Situationen sein Verhalten entsprechend anpassen.

Die Verhaltensweisen, die vom Lerner gezeigt werden, unterscheidet Skinner in respondentes Verhalten und in operantes Verhalten. Beim respondenten Verhalten reagieren die Lernenden auf einen vorher erfolgten Stimulus. Dies entspricht auch dem Modell des klassischen Behaviorismus. Das operante Verhalten hingegen wird spontan geäußert. Hierbei steht die Reaktion zunächst in keiner Verbindung zu einem vorherigen Stimulus. Erst auf die Reaktion der Umwelt auf das geäußerte Verhalten erfolgen zukünftige Konsequenzen. Bei positiven Reaktionen (z. B. Lob) wird das Verhalten beibehalten (gelernt), bei negativen Reaktionen (z. B. Strafe) wird das Verhalten zukünftig unterlassen.

Ein Mittel, diesen Lernprozess zu unterstützen, ist die so genannte Verstärkung. Ein Verstärker ist jede Handlungskonsequenz, die sich steuernd auf das gezeigte operante Verhalten auswirkt und die Wahrscheinlichkeit erhöht, dass dieses Verhalten in der Zukunft wiederholt wird.

Beispiele für behavioristische Lernsysteme

Folgende Lernsysteme folgen der behavioristischen Theorie: Drill & Practise, Präsentationen, guided tours, multiple choice, Lernmaschinen, Programmierte Instruktionen.

2.3.2 Kognitivismus

Der Kognitivismus wurde durch Theorien von Piaget [Pia77] und Brunner [Bru66] begründet. Der Lernende wird als Individuum begriffen, das aktiv die äußeren Reize verarbeitet und nicht durch diese steuerbar ist. Es wird davon ausgegangen, dass Lernen auf kognitiven Strukturen basiert und der Lernende Eindrücke durch seinen Erfahrungs- und Entwicklungsstand selektiv wahrnimmt, interpretiert und verarbeitet. Nach dieser Theorie gewinnt die Relation von externen Bedingungen des Lernens (z. B. Lernprogramme, Applets, Gruppentools) mit den internen Bedingungen (Erfahrungs- und Entwicklungsstand) an Bedeutung. Die Wahrnehmungspsychologie liefert in diesem Zusammenhang wesentliche Erkenntnisse zur positiven Beeinflussung (z. B. durch farbliche Gestaltung der Oberfläche).

2.3.3 Konstruktivismus

Bei diesem Ansatz geht man davon aus, dass der Lernende auf Basis seiner subjektiven Erfahrungsstrukturen die "Wirklichkeit" individuell konstruiert [MV87]. Der Konstruktivismus lehnt den im Kognitivismus enthaltenen Objektivismus ab (siehe auch [Jon91]). Nach dem konstruktivistischen Ansatz ist Wissen nicht ein Abbild der externen Realität, sondern eine Funktion von Erkenntnisprozessen. Dies hat zur Folge, dass Lernmaterialien als Informations- und Werkzeugangebote für selbstgestaltete Lernprozesse zu konzipieren sind. Diese können jedoch den Lernprozess selbst nicht steuern [Eul94]. Die Realisierung findet meist in Lernumwelten statt. Hierbei gibt es drei Ansätze:

1. Cognitive Tools:

Kognitive Werkzeuge bieten den Lernenden Gelegenheit, eigene Konzepte zu generieren und umzusetzen. Die Lernumwelt erfährt damit eine Modellierung durch den Lerner [Koz87].

2 Lernen durch Üben

2. Cognitive Apprenticeship:

Der Lernvorgang wird als Lehrlingsverhältnis konzipiert, wobei die Übung als "Coaching" unter Beratung des Meisters stattfindet, der sich mit fortschreitender Erfahrung des Lehrlings immer mehr zurücknimmt [CBN89].

3. Knowledge Communities:

Das Lernen wird als kommunikatives Handeln in Wissensgemeinschaften verstanden. Die Konstruktion von Realität erfolgt auch in der Auseinandersetzung mit der sozialen Umwelt. Die Kommunikation (z. B. innerhalb von Netzwerken) sollte deshalb bei Lernprozessen unterstützt werden [Bro85].

Sozialer Konstruktivismus

Der soziale Konstruktivismus geht davon aus, dass die Wirklichkeit durch menschliche Kommunikation gesellschaftlich konstruiert und damit auch interpretiert wird. Hierbei ist die Konstruktion historisch und veränderbar. Als primäres Medium dient hierbei die Sprache. Somit bringen soziale Konstruktionen (Vorstellungen, Vorurteile, Urteile und Meinungen) wirklichkeitsschaffende Erkenntnisse hervor. Damit wird Wirklichkeit hergestellt, indem auf eine bestimmte Art und Weise über sie gesprochen wird und dabei vielfältige Beziehungen koordiniert werden [BTL80], [HB00].

2.4 Modelle zur Wissensvermittlung

Aufbauend auf den verschiedenen Lerntheorien haben sich Modelle zur Wissensvermittlung gebildet. Zwei bekannte Modelle sollen nun vorgestellt werden: Der Learning Cycle von Mayes et al. [MCTR94] sowie die Wissensspirale von Nonaka und Takeuchi [NT97].

2.4.1 Der Learning Cycle nach Mayes

Mayes, Coventry, Thomson und Mason [MCTR94] schlugen insbesondere für mediengestützte Kurse den so genannten Learning Cycle vor. Abbildung 2.2 stellt das Modell grafisch dar. Es werden die drei Phasen eines idealtypischen Lernprozesses beschrieben: Konzeptualisierung, Konstruktion und Dialog. Mit jeder Phase soll eine

tieferer Wissensverarbeitung erreicht werden. Die drei Phasen werden zyklisch beschrieben, so dass sie innerhalb einer Übung mehrfach durchlaufen werden können.

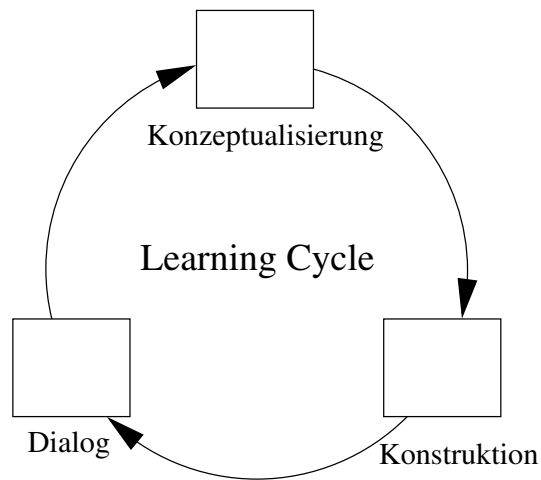


Abbildung 2.2: Der Learning Cycle nach [MCTR94]

Konzeptualisierung

In der ersten Phase, der Konzeptualisierung, wird der Lerner mit einem neuen Wissensgebiet konfrontiert. Der Lernende orientiert sich zunächst an relevanten Lerninhalten und exploriert diese mit dem Ziel einer ersten Interpretation des Lernstoffs. Ziel ist es, dass er sich ein kognitives Konzept entwickeln kann, um die Anknüpfung zusätzlicher Lerninhalte zu ermöglichen. Das Faktenwissen, welches in dieser Phase erworben wird, ist noch sehr oberflächlich.

Konstruktionsphase

In der zweiten Phase, der Konstruktionsphase, interpretiert der Lerner neue Informationen und verknüpft sie mit seinem Vorwissen. Haupttätigkeiten dieser Phase sind die Selektion und die Rekombination. In dieser Phase ist es entscheidend, den Schwierigkeitsgrad der Aufgaben an die Fähigkeiten der Studierenden anzupassen.

Dialog

Die dritte Phase, der Dialog, hat zum Ziel, eine vollständige Repräsentation der gelernten Inhalte zu erreichen. Dies erfolgt mit dem Mittel der Externalisierung des bisher Erlernten. In den meisten Fällen geschieht dies durch eine fachliche Diskussion, die der Lernende mit sich selbst oder mit anderen Menschen durchführt.

2.4.2 Vermittlung von Wissensinhalten durch das dynamische Modell nach Nonaka und Takeuchi

Basierend auf dem in der Psychologie bekannten ACT (adaptive control of thought)-Modell (nach Anderson [And83], Singley und Anderson [SA89]) entwickelten Nonaka und Takeuchi ihr Modell. Nach dem ACT-Modell können sich kognitive Fähigkeiten nur dann entwickeln, wenn das gesamte deklarative Wissen - entsprechend dem expliziten Wissen - in prozessuales - implizites - Wissen übergeht. Beispielsweise gilt dies bei dem Erlernen des Radfahrens oder auch des Klavierspielens. Das Modell ist jedoch dadurch begrenzt, dass es sich ausschließlich auf den Erwerb und Austausch von prozessualem (implizitem) Wissen konzentriert. Es gäbe demnach nur einen einseitigen Prozess der Wissensumwandlung vom deklarativen zum prozessualen Wissen. Das Modell von Nonaka hat jedoch einen interaktiven Spiralprozess zum Gegenstand.

Das dynamische Modell nach [NT97][S. 73 ff.] fußt auf der Prämisse, dass durch eine Interaktion zwischen dem expliziten und impliziten Wissen weiteres Wissen geschaffen sowie vorhandenes Wissen erweitert wird. Dieser Prozess wird von Nonaka als Wissensumwandlung bezeichnet. In Abbildung 2.3 werden zunächst die vier Formen der Wissensumwandlung dargestellt:

1. Sozialisation

Die Sozialisation ist ein Erfahrungsaustausch, bei dem implizites Wissen entsteht. Der Austausch erfolgt z. B. durch gemeinsame mentale Modelle und technische Fertigkeiten. Dies ist unter anderem der Fall beim Lernen durch Beobachtung, Nachahmung oder Praxis [NT97][S. 75]. Notwendig für den Erwerb und Austausch des impliziten Wissens ist jedoch eine gemeinsame Erfahrung. Zur Realisierung der Sozialisation als Lernform bieten sich folgende klassische Maßnahmen an:

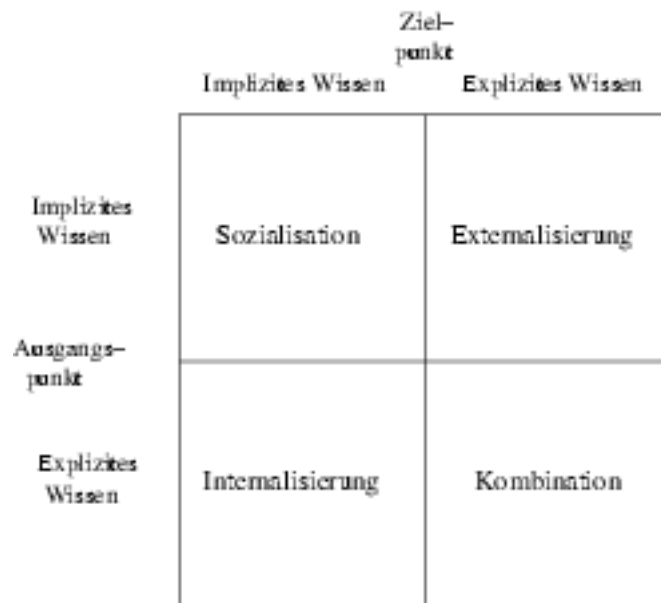


Abbildung 2.3: Modell zur Wissensumwandlung nach [NT97][S. 75]

- Brainstorming Camps

Hierzu werden Workshops außerhalb der gewohnten Arbeitsumgebung abgehalten. Sie sind offen zu handhaben; dies bedeutet, dass auch andere interessierte Personen teilnehmen sollen. In der Diskussion spielen Status und Qualifikation keine Rolle. Während des gesamten Workshops ist eine eiserne Regel einzuhalten: keine Kritik ohne konstruktive Vorschläge.

- In die Lehre gehen

Darunter versteht man die angeleitete Beobachtung, Nachahmung und Praxis.

2. Externalisierung

Unter Externalisierung verstehen Nonaka und Takeuchi den Prozess der Artikulation vom impliziten Wissen in expliziten Konzepten. Zunächst ist hierzu das implizite Wissen in Form von Metaphern, Analogien, Modellen und Hypothesen darzustellen. Um die damit verbundenen Ungenauigkeiten auszugleichen, ist eine Interaktion erforderlich. Diese wird meist durch einen Dialog und eine kollektive Reflexion erreicht (wie es meist bei Konzepten passiert). Hierbei werden die Prinzipien der Induktion und Deduktion miteinander verknüpft [NT97][S. 77].

2 Lernen durch Üben

3. Kombination

Die Kombination ist ein Prozess der Erfassung von Konzepten innerhalb eines Wissenskomplexes und dient dazu, verschiedene Bereiche von explizitem Wissen miteinander zu verbinden. Der Austausch geschieht mit Hilfe von Medien. Auch durch explizites Wissen kann neues Wissen generiert werden. Dies geschieht z. B. durch Sortieren, Hinzufügen, Kombinieren oder Klassifizieren von explizitem Wissen [NT97][S. 81].

4. Internalisierung

Internalisierung ist ein Prozess zur Eingliederung expliziten Wissens in das implizite Wissen. Dieser Prozess ist eng verwandt mit dem so genannten "learning by doing". Wenn Erfahrungen durch Sozialisation, Externalisierung und Kombination in Form von gemeinsamen mentalen Modellen oder technischen Know-how internalisiert werden, werden sie zu einem wertvollen Wissenskapital. Unterstützt werden kann der Prozess durch die Vermittlung des impliziten Wissens auf Medien, indem eigene Erfahrungen geschildert werden. Zum Beispiel könnte die Schilderung von eigenen Erfolgserlebnissen dazu führen, dass sich die Empfänger ein eigenes implizites Bild machen.

Voraussetzungen zur Wissensspirale

Um diesen Prozess zu initiieren, sind nach Nonaka und Takeuchi [NT97][S. 88 ff.] insbesondere folgende Voraussetzungen zu erfüllen:

- Intention
Es muss erklärtes Ziel sein, dass Wissen vermittelt und geschaffen werden soll.
- Autonomie
Auf individueller Ebene sollen alle Teilnehmer so autonom handeln können, wie es die Umstände erlauben. Dies führt nicht nur zu unerwarteten Chancen, sondern fördert auch die Motivation zur Schaffung neuen Wissens. Organisatorische Grundlage hierfür bildet ein bekanntes Phänomen, das "autopoietische System". Hierbei erfolgt die Steuerung in komplexen Organismen nicht durch eine hierarchische Struktur, sondern jedes Organ kontrolliert sich selbst.
- Fluktuation und kreatives Chaos
Hierunter ist insbesondere die Anregung zu verstehen, eine Wechselwirkung der Einrichtung mit der Umwelt herbeizuführen. Unter Fluktuation ist hierbei

eine "nichtrekursive Ordnung" zu verstehen, deren Muster am Anfang schwer vorhersagbar ist. Dies führt unter Umständen zu einem Zusammenbruch von Routineabläufen, Gewohnheiten oder kognitiven Bezugssystemen. Ein Zusammenbruch zwingt hierbei zum Dialog als Mittel sozialer Interaktion, um neue Konzepte zu finden.

- Redundanz

Hierbei versteht man unter Redundanz die Existenz von Informationen, die über die unmittelbaren operativen Bedürfnisse der Teilnehmer (Studierende) hinausgehen; ein absichtliches Überschneiden von Informationen über einzelne Themengebiete hinweg. Es ist hierbei allerdings auf eine Balance zwischen verminderter Effizienz durch Informationsüberflutung und der Schaffung bzw. dem Besitz von Informationen zu achten.

- Notwendige Vielfalt

Die interne Vielfalt einer Organisation soll der Komplexität des Umfeldes entsprechen, um dessen Anforderungen gerecht zu werden [NT97][S. 98]. Möglichkeiten zur Realisierung sind der gleichberechtigte Zugang zu Informationen sowie der häufige Wandel der Organisationsstruktur.

Innovation durch die Wissensspirale

Nach Nonaka und Takeuchi ergibt sich eine echte Innovation erst dann, wenn explizites und implizites Wissen zusammenwirken. Dies erfordert eine dynamische Interaktion und Verschiebung der verschiedenen Wissensumwandlungsprozesse, wie sie in Abbildung 2.4 dargestellt wird [NT97][S. 84].

Die Sozialisation geht meist vom Aufbau eines Interaktionsfeldes aus, welches die Weitergabe von Erfahrungen und mentalen Modellen erleichtert.

Die Auslösung der Externalisierung erfolgt entweder durch einen konstruktiven Dialog oder durch kollektive Reflexion, die mit Hilfe von Metaphern und Analogien die Artikulation von ansonsten nur schwer mitteilbaren impliziten Kenntnissen führt. Die Kombination entsteht durch die Verbindung neu geschaffenen und bestehenden Wissens aus anderen Bereichen, um sie in einer neuen Idee zu verschmelzen. Internalisierung resultiert schließlich aus dem "Learning by doing".

Hierzu ist zunächst das vorhandene Wissen zu mobilisieren. Durch die vier Formen der Umwandlung wird es verstärkt und dringt dadurch in eine höhere ontologische

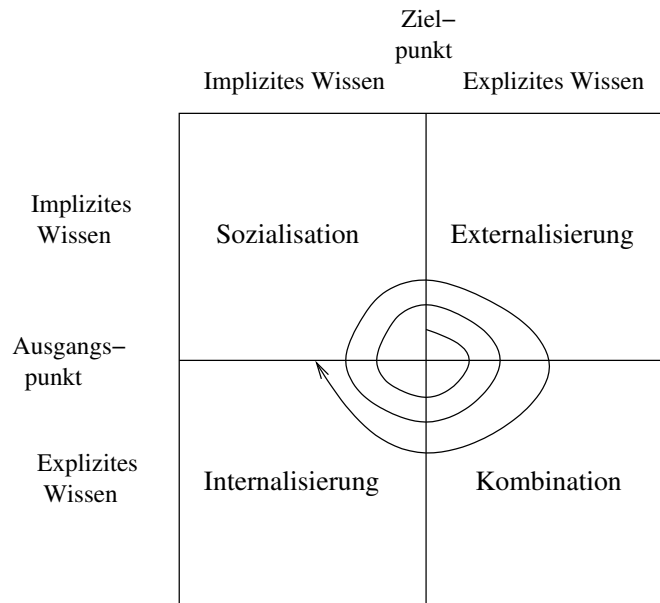


Abbildung 2.4: Modell zur Wissensumwandlung nach [NT97][S. 84]

Schicht vor. Dieser Vorgang wird von Nonaka und Takeuchi als *Wissensspirale* bezeichnet [NT97][S. 86] und ist in Abbildung 2.5 schematisch dargestellt. Die Wissensschaffung ist somit ein Spiralprozess, der ausgehend von der individuellen Ebene immer mehr Interaktionsgemeinschaften umfasst und die Grenzen von Gruppen, Lehrstühlen, Instituten und sogar Universitäten überschreitet.

Das Modell von Nonaka und Takeuchi erscheint insbesondere für die Wissensvermittlung in großen Einrichtungen wie z. B. Universitäten besonders geeignet. Deshalb wurde beschlossen, dieses Modell als lerntheoretische Fundierung des CATS-Systems zu verwenden.

2.5 Bedeutung von Übungen in der Lehre

Um eine verbesserte Wissensvermittlung zu erreichen, führen fast alle Bildungseinrichtungen Veranstaltungen durch, die die Studierenden dazu bewegen sollen, sich selbstständig mit dem Stoffgebiet auseinanderzusetzen. Zudem soll eine Interaktion zwischen Dozierenden und Lernenden angeregt werden. Hierbei werden den Studierenden kleine überschaubare Aufgaben gestellt, die sie selbstständig zu lösen haben. Eine solche Veranstaltung wird auch als Übung bezeichnet.

In der klassischen Lehre sind Präsenzübungen integraler Bestandteil der Lehre. Sie

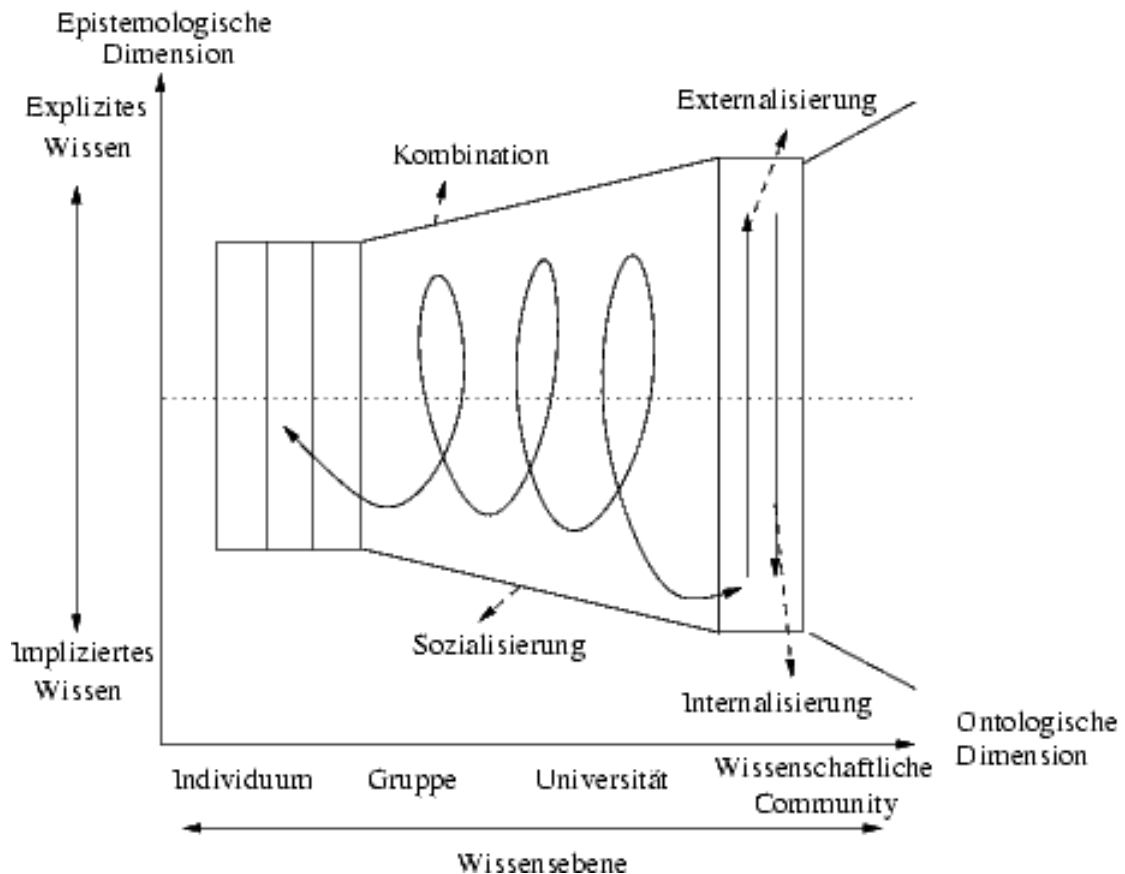


Abbildung 2.5: Spirale der Wissensschaffung in der Universität nach [NT97][S. 87]

dienen verschiedenen Zwecken:

- Internalisierung des Erlernten durch "Learning-by-doing"
- Die Studierenden erhalten eine Rückmeldung über ihren individuellen Leistungsstand.
- Die Lehrenden erhalten die Information über den Leistungsstand der Studierenden.
- Die Arbeit zusammen mit Kommilitonen ist motivierend und fördert das Lernen.

Diese Vorteile von Übungen wurden in einigen Untersuchungen belegt: Morrison et al. [Mor95], Haynie [Hay94], Hamaker [Ham86].

2.5.1 Pädagogische Motivation der Gruppenarbeit

Eine Ausprägung der Übung besteht in der Gruppenarbeit. Diese spielt in der klassischen Ausbildung, quer durch alle Ebenen des Bildungssystems, eine wichtige Rolle. Die Studierenden müssen selbst aktiv werden, die Lehrer erhalten eine umfassende Rückmeldung bezüglich des Leistungsstandes der Studierenden, und schließlich wird Gruppenarbeit allgemein als sehr motivierend empfunden. Wie in Abschnitt 2.4 dargestellt, haben sehr viele theoretische Modelle des menschlichen Lernens eine Phase, in der die Lerner Wissensinhalte mit anderen diskutieren und damit reflektieren sollen. Durch diesen Vorgang wird das Wissen gefestigt, die persönlichen Problemlösungsfähigkeiten werden verbessert, das Erinnerungsvermögen an Fakten wird verstärkt, und schließlich erfährt die Gruppenarbeit eine hohe Akzeptanz bei den Studierenden, wie Untersuchungen belegen [KPCM02], [BBC00].

Es ist daraus zu folgern, dass die einfache, jederzeit mögliche Kommunikation zwischen allen Lernern und zwischen Lernern und Tutoren eine bedeutende Voraussetzung für erfolgreiche Übungen ist. Demgegenüber werden Studierende, die für sich alleine lernen, keine optimalen Lernerfolge erzielen.

Erkenntnisse aus der Pädagogischen Psychologie

Slavin [Sla80], [Sla93] bildet eine "Zauberformel" für erfolgreiches kooperatives Lernen. Demnach ist die Kooperation an bestimmte Bedingungen geknüpft:

- Definition eines Gruppenzieles
Dies bedeutet, dass jeder Teilnehmer sich mit dem Ziel der Gruppe identifizieren kann und dieses dann auch verfolgt.
- Individuelle Verantwortlichkeit
Der individuelle Beitrag zum Gruppenerfolg sollte nachvollziehbar und für alle Gruppenmitglieder erkennbar sein (Erkennbarkeit der Einzelleistung).
- Echte Gruppenaufgaben
Die gestellte Aufgabe soll nur in Form einer Gruppenarbeit überhaupt lösbar sein.
- Ressourceninterdependenz
Dies bedeutet, dass es ein Abhängigkeitsverhältnis der Gruppenmitglieder un-

tereinander gibt. Die Mitglieder sind auf die Fähigkeiten und Kenntnisse der anderen Mitglieder angewiesen.

- Offene Lösung
Die Lösung der Gruppenaufgabe soll nicht bereits im Vorhinein festgelegt oder offensichtlich sein.
- Inhaltlich motivierend (spannende Aufgabenstellung)
Schließlich soll die Aufgabenstellung selbst eine interessante Fragestellung behandeln und dadurch intrinsisch motivieren.

Wenn die Motivation zwar vorhanden ist, aber trotzdem die fachlichen Fähigkeiten fehlen, besteht die Gefahr einer inadäquaten Arbeitsteilung und der Bildung eines "secret masterplan" durch die Meinungsführer. Um dem entgegenzuwirken, also den "secret masterplan" in einen "open masterplan" zu transformieren, empfiehlt sich das Mittel des Skripting. Dies bedeutet, dass genaue Verhaltensnormen und Verfahren vorher eindeutig festgelegt werden müssen. Insbesondere ist hierbei auch auf eine Sequenzierung der Aufgaben zu achten, also darauf, dass die Aufgaben in einer pädagogisch sinnvollen Reihenfolge zu bearbeiten sind.

Heckhausens Motivationsmodell der individuellen Handlungsentscheidung

Bei der Gestaltung der Gruppenaufgaben sind insbesondere Erkenntnisse aus der Motivationsforschung zu berücksichtigen. Heckhausen [Hec89] formulierte in seinem Modell vier Fragen, die die individuelle Handlungsentscheidung erklären:

1. Erscheint das Ergebnis bereits durch die Situation festgelegt?
Zum Beispiel: Genügt die formale Anmeldung, um einen Leistungsnachweis zu erhalten?
2. Kann man das beabsichtigte Ergebnis selbst herbeiführen?
Zum Beispiel: Hat mein individueller Beitrag einen Mehrwert für die Gruppenarbeit?
3. Sind einem die möglichen Folgen des Ergebnisses wichtig genug?
Lohnt sich der Erwerb dieses Leistungsnachweises für mich? Ist er z. B. in der Studienordnung vorgeschrieben?

4. Zieht das Ergebnis auch die gewünschten Folgen nach sich?

Zum Beispiel: Wird der erbrachte Leistungsnachweis vom Prüfungsamt anerkannt?

Extrinsische Ansätze zur Motivation nach Slavin

Slavin [Sla80], [Sla93] schlägt zur weiteren Motivation der einzelnen Studierenden auch extrinsische Ansätze vor. Demnach sind Lernumgebungen so zu gestalten, dass die individuelle Leistung erkennbar bleibt, eine Belohnung (z. B. Benotung) aber für alle Gruppenmitglieder gleich ist. Dieser Punkt ist jedoch häufig aufgrund der rechtlichen Rahmenbedingungen, wonach die individuellen Beiträge bewertet werden müssen, nur sehr eingeschränkt realisierbar.

2.5.2 Ausgestaltung von Übungen in der Präsenzlehre

Um eine verbesserte Wissensvermittlung zu erreichen, finden in der Präsenzlehre Präsenzübungen statt. Sie sind in aller Regel als größere Veranstaltungen (ca. 20 Studierende) zu regelmäßigen Terminen (in aller Regel einmal pro Woche) konzipiert. Um diese Übungen effizient zu gestalten, werden in den meisten Fällen Übungsaufgaben im Voraus ausgegeben. Die Studierenden sind aufgefordert, sich in Heimarbeit mit dem Stoff auseinanderzusetzen und einen Lösungsvorschlag selbständig zu erarbeiten. Innerhalb der Präsenzveranstaltung werden dann im Idealfall interaktiv die Lösungsvorschläge der Studenten besprochen und gemeinsam mit einem Tutor eine optimierte Lösung erarbeitet. Hierbei sollen insbesondere Erfahrungswerte an die Studierenden, die sich in aller Regel zum ersten Mal mit den Inhalten auseinandergesetzt haben, vermittelt werden. Dies können Verfahren, Vereinfachungen oder auch Kontrollmöglichkeiten sein.

Da sich während eines vorher bestimmten Zeitraumes eine große Anzahl an Studierenden in einem gemeinsamen Raum befindet, ergibt sich neben dem fachlichen Austausch auch eine soziale Interaktion. Diese führt im besten Fall auch zur Gründung von Lerngruppen. Zum Teil werden diese Vorgänge durch die Übungskonzeption gefördert. Aus diesem Grund gibt es spezielle Mentorenprogramme, oder es werden explizit Gruppenaufgaben gestellt.

Oft erweisen sich solche Lerngruppen als extrem stabil und halten, falls sie bereits im ersten Semester gegründet werden, während des gesamten Studiums und zum Teil darüber hinaus.

2.5.3 Übungen in der Fernlehre

In der Fernlehre ist das Szenario sehr verschieden. Aufgrund der räumlichen Distanz sind klassische Face-to-Face Übungsveranstaltungen nicht durchführbar. Aufgrund der asynchronen Ausrichtung der meisten Lehrangebote besteht meist keine Möglichkeit, in einer größeren Gruppe regelmäßig zusammenzukommen. Somit werden in aller Regel Übungsaufgaben an die Studierenden distribuiert, und diese sind aufgefordert, ihre Lösungen zu festen Zeitpunkten an die Institution zurückzuschicken. Dort werden die Aufgaben überprüft, korrigiert und mitsamt einer Musterlösung an die Studierenden zurückgeschickt.

Eine enge Rückkopplung und Diskussion, wie aus der Präsenzlehre bekannt, findet nicht statt.

Aufbau des Fernlehrszenarios

Das Fernlehrszenario stellt sich meist wie folgt dar: Eine zentrale Institution (z. B. eine Fernuniversität) übernimmt die zentrale Administration aller Verwaltungsvorgänge und sorgt für die logistischen Prozesse der Fernlehre (Erstellung der Kursmaterialien, Versendung von Kursmaterialien, Abrechnung etc.). Die Studierenden arbeiten als Fernstudierende in aller Regel räumlich und zeitlich verteilt. In manchen Szenarien (z. B. bei der FernUniversität in Hagen) kann es noch eine Zwischenstufe in Form von Studienzentren geben. Dort werden in regelmäßigen Abständen Tutorien abgehalten. Die Fernlehre erhält hierdurch eine Präsenz-Unterstützung.

Zeit-Divergenz

In der telemedialen Lehre wird zwischen synchroner und asynchroner Kommunikation unterschieden. Die synchronen Szenarien sind dadurch gekennzeichnet, dass alle Teilnehmer gleichzeitig zueinander in Verbindung stehen. Die Lerner sind zur gleichen Zeit über ein Medium erreichbar. Typische Szenarien sind so genannte Televorlesungen. Hierbei wird eine Vorlesung zeitgleich zu einem oder mehreren Orten übertragen. Die Übertragung enthält neben Audio und Video auch die Übertragung der Präsentationunterlagen inkl. etwaiger Annotationen.

Eine weitere Unterscheidung betrifft die räumliche Verteilung. Hierbei können die Studierenden an einem Ort oder an verschiedenen Orten aktiv sein.

Abbildung 2.6 stellt die Einordnung von synchronen und asynchronen Szenarien und die damit verbundenen Methoden in einer Übersicht dar. Je nach Konfiguration

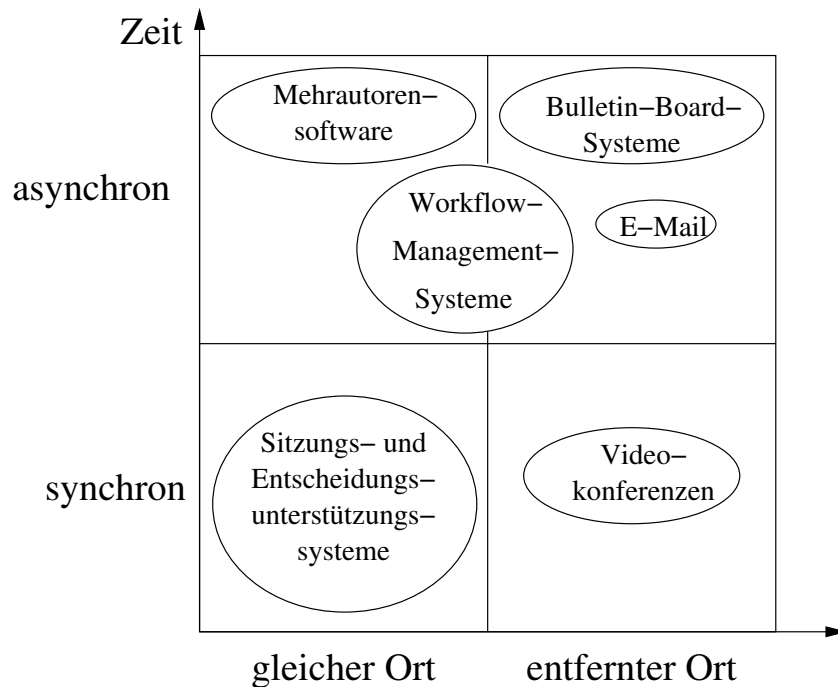


Abbildung 2.6: CSCL nach Raum/Zeit nach [Mit98][S. 22]

sind unterschiedliche Hilfsmittel zur Unterstützung der Lerner notwendig. Diese reichen von einfachen Kommunikationsmitteln (z. B. E-Mail) bis hin zu komplexen Systemen, wie sie z. B. Lernplattformen darstellen.

Beschränkte Kommunikation

Da die Studierenden im Fernlehrmodus räumlich verteilt sind, reduziert sich ihre Kommunikation. Allen hat in Untersuchungen bezüglich des Kommunikationsverhaltens von Arbeitnehmern in Büroräumen festgestellt, dass bereits dann, wenn die räumliche Trennung nur wenige Meter (z. B. ein Teil einer Arbeitsgruppe arbeitet in einem anderen Stockwerk) beträgt, nicht nur die direkte "Face-to-Face"-Kontaktaufnahme reduziert wird, sondern auch die elektronisch vermittelte (z. B. per Telefon) sehr stark reduziert wird [All77], [All97]. Es ist daher festzuhalten, dass eine elektronische Kommunikationstechnologie für sich alleine nicht ausreicht, um die Kontaktaufnahme zu fördern. Vielmehr müssen noch weitere Anforderungen hinzutreten. Dieser Sachverhalt ist insbesondere für Fernstudierende verhängnisvoll, da gerade die im vorherigen Abschnitt vorgestellten Modelle der Wissensvermittlung eine intensive Kommunikation voraussetzen. Die alleinige Verfügbarkeit einer elektroni-

schen Kommunikationsmöglichkeit reicht auch nicht aus, um die Kontaktaufnahme zwischen den Studierenden zu initiieren.

Geringe soziale Interaktion

In der Folge des Kommunikationsabbrisses findet kaum noch eine soziale Interaktion statt, im Gegensatz zu Präsenzstudierenden. Im Fernstudium gibt es sehr hohe Abbrecherzahlen. Diese reichen an die 80-98 % (je nach Studienfach) heran [Fri04a]. Wenn auch die hauptsächliche Begründung der geringen sozialen Integration als Ursache für einen Abbruch des Fernstudiums umstritten ist, so ist es dennoch unzweifelhaft einer der bestimmenden Gründe.

Existierende Lösungsansätze zur Überwindung der zeitlichen und räumlichen Distanz

Die bekannten Fernuniversitäten haben diese Problematik erkannt und versuchen, diese durch verschiedene Maßnahmen zu kompensieren. Diese sind im Einzelnen:

Spezielle mentorielle Betreuung Die Teleakademie der FH Furtwangen bildet zu diesem Zweck spezielle Fachkräfte weiter und sieht die Mentoren als Trainer, die individuell die Studierenden anleiten und motivieren. Die Mentoren fordern regelmäßig von den Studierenden Leistungen ein.

Kursspezifische telefonische Sprechstunden Diese Art der Unterstützung ist jedoch nur bei einer sehr guten Betreuungsrelation durchführbar, wie sie lediglich bei kleinen und mittleren Veranstaltungen vorhanden ist. Insbesondere bei Massenveranstaltungen mit an die tausend Studierenden (z. B. Einführung in die BWL an der FernUniversität in Hagen) ist dies sehr problematisch, da die Betreuungskapazitäten fehlen.

Präsenzübungen in Studienzentren Diese Veranstaltungsart durchbricht das Paradigma der reinen Fernlehre. Die räumliche Distanz wird überwunden, und entsprechende Veranstaltungen können vor Ort angeboten werden. Dies ist allerdings sehr kostenintensiv (mehrfaches Personal, Räumlichkeiten). Zudem ist eine flächendeckende Versorgung mit Studienzentren nicht gewährleistet. Insbesondere ausländische Studierende können in aller Regel diese Maßnahme nicht nutzen.

Präsenz-Blockveranstaltung Durch diese Maßnahme wird ebenfalls das Prinzip der Fernlehre durchbrochen. Die Veranstaltung findet meist an einem zentralen Ort statt. Hierbei hat allerdings die Veranstaltung aufgrund der hohen Anzahl an Teilnehmern und der komprimierten Stoffvermittlung eher den Charakter einer Vorlesung mit wenig Interaktionsmöglichkeiten.

Kursspezifische Kontaktlisten Die FernUniversität in Hagen z. B. bietet den Studierenden, nach entsprechender freiwilliger Freigabe ihrer Kontaktdaten, so genannte Kontaktlisten an. Diese sind nach Regionen geordnet und enthalten die Adressen der Beleger des jeweiligen Kurses. Dieses Instrument wird jedoch kaum genutzt, da die Hemmschwelle, einen "anonymen" Kommilitonen anzurufen, doch sehr hoch ist.

Internet News-Foren An der FernUniversität in Hagen gibt es für sehr viele Kurse entsprechende News-Foren. Allerdings muss, damit diese Form des Austausches funktioniert, eine große Anzahl an aktiven Belegern vorhanden sein, und es sollte sich eine Diskussionskultur ausbilden, was aufgrund der hohen Fluktuation in den einzelnen Kursen schwierig ist. Da zudem nur eingeschriebene Studierende Zugang zu den Foren haben, ist die Kommunikation auf die Mitglieder der FernUniversität in Hagen beschränkt.

Chat-Räume In einigen Versuchen wurden Chat-Räume für bestimmte Kurse an der FernUniversität in Hagen eingerichtet. Die Beteiligung der Studierenden war jedoch sehr gering, so dass dieses Angebot stark reduziert wurde.

2.5.4 Lernziele im universitären Umfeld

Neben dem Problem der räumlichen und zeitlichen Trennung kommen gerade in der universitären Ausbildung noch sehr hohe Lernziele hinzu, die u. a. in §7 HRG definiert sind: "Lehre und Studium sollen den Studenten auf ein berufliches Tätigkeitsfeld vorbereiten und ihm die dafür erforderlichen fachlichen Kenntnisse, Fähigkeiten und Methoden dem jeweiligen Studiengang entsprechend so vermitteln, dass er zu wissenschaftlicher oder künstlerischer Arbeit und zu verantwortlichem Handeln in einem freiheitlichen, demokratischen und sozialen Rechtsstaat befähigt wird." Um zu wissenschaftlicher Arbeit befähigt zu sein, müssen die Studierenden neben dem aktuellen Fachwissen zudem in der Lage sein, durch methodisch geordnetes und kritisch reflektierendes Denken wahre Erkenntnisse zu ermitteln.

2.5 Bedeutung von Übungen in der Lehre

Gerade das Erlernen des kritisch reflektierenden Denkens setzt eine intensive Kommunikation mit anderen voraus, die im Fernlehrszenario so nicht gegeben ist.

Die einzelnen Methoden, die die Studierenden erlernen sollen, sind zudem von Fach zu Fach sehr unterschiedlich. In den Naturwissenschaften etwa sind die einzelnen Verfahren oft durch Experiment, Modellbildung und Verifikation gekennzeichnet, während zum Beispiel in der Rechtswissenschaft die Dogmatik und die Auslegung eine wesentliche Rolle spielen.

3 Existierende Vorarbeiten

Der Einsatz von Computern zur Unterstützung des Lernens begann bereits in den 50er Jahren mit einfachen Lernmaschinen. Später erfolgte der Einsatz in Form von Animationen, Simulationen und automatisierten Übungssystemen.

Mit Hilfe der Künstlichen Intelligenz (KI) wurden so genannte Intelligente Tutorielle Systeme (ITS) entwickelt, die komplette Wissensdomänen abbilden konnten. Man versuchte dadurch, eine Optimierung der Lernprozesse bei den einzelnen Studierenden zu erreichen. Schließlich nutzte man auch die immer weiter verbreiteten Rechneternetze dazu, Systeme zu entwickeln, die das kooperative Lernen unterstützen. Dies bezeichnet man auch als Computer Supported Cooperative Learning (CSCL). Parallel zu dieser Entwicklung wurden immer komplexere Lernumgebungen entwickelt, die insbesondere das entdeckende Lernen im Sinne des Konstruktivismus bei den Studierenden fördern sollen. Zusätzlich bieten Lernplattformen eine Unterstützung in der organisatorischen Abwicklung von Kursen.

In diesem Kapitel werden exemplarisch frühere Arbeiten genannt, die in einem Zusammenhang mit dem entwickelten System stehen.

3.1 Animationen und Simulationen

Animationen und Simulationen stellen eine sehr mächtige Möglichkeit dar, Erkenntnisprozesse beim Lerner zu unterstützen. Hierbei werden bekannte Metaphern zur Visualisierung verwendet und durch ein entsprechend erstelltes Modell der Realität abgebildet. Sie dienen der Vermittlung des Lehrstoffs und können insbesondere bei der Präsentation von Übungsaufgaben eine wichtige Rolle spielen.

3.1.1 Animationen

Animationen dienen der Veranschaulichung von zeitabhängigen Prozessen. Hierbei verwenden Animationen in der Regel allgemein bekannte oder fachspezifische Me-

3 Existierende Vorarbeiten

taphern zur Visualisierung. Für die Programmierung von einfachen Animationen sind heute kaum noch spezielle Kenntnisse einer Programmiersprache erforderlich. Die Erstellung wird bereits von den meisten Präsentationsprogrammen (z. B. Microsoft Powerpoint) unterstützt. Mächtigere, insbesondere fachspezifische Animationen sind im Internet für fast jedes Wissensgebiet verfügbar.

Reine Animationen sind eigengesteuert, die Interaktion des Benutzers, insbesondere Änderungen der Modellparameter, ist dem Nutzer in aller Regel nicht möglich. Dafür ist die Erstellung einer Animation andererseits auch einfacher, da nicht eine ganze Systemwelt modelliert werden muss.

Ein Projekt, welches interaktive Animationen zum Gegenstand hat, wird von von Burger, Mecklenburg und Rothermel in [Cla98][S. 176 ff.] beschrieben. Hierbei handelt es sich um interaktive Animationen, welche Kommunikationsprotokolle auf generischem Wege darstellen.

Auch am Lehrstuhl Praktische Informatik IV, an dem diese Arbeit entstand, wurde im Zeitraum von sieben Jahren eine umfangreiche Sammlung von Animationen erstellt. Diese wurden in aller Regel in der Programmiersprache Java in Form von Applets programmiert.

In einem weiteren Schritt wurden im Rahmen von verschiedenen Projekten interaktive Applets entwickelt, die im Sinne des Kognitivismus dem Studierenden ein selbstgesteuertes Lernen anboten. Hierzu wurden insbesondere verstärkt Interaktionsmöglichkeiten geschaffen, die es dem Lerner ermöglichen Einfluss auf das Verhalten der Animation zu nehmen. So wurde z. B. die ein- wie die zweidimensionale diskrete Kosinus Transformation DCT als Applet entwickelt [Kuh01][Sch01]. Die DCT transformiert ein im Ortsraum gegebenes Eingangssignal in den Frequenzraum. Wobei die Repräsentation des Signal durch entsprechend diskrete Koeffizienten erreicht wird. Diese Funktion ist insbesondere für moderne Verfahren in der Signalverarbeitung von hoher Bedeutung und daher auch Gegenstand entsprechender Lehrveranstaltungen. Die entsprechenden Algorithmen sind jedoch zum Teil sehr komplex und stoßen bei den Studierenden oftmals auf Schwierigkeiten. Diese Probleme konnten jedoch durch das entsprechende Applet vermindert werden. In einer umfangreichen Evaluierung konnte, falls die Anwendung der Animationen geskriptet wird, eine signifikante, positive Lernwirksamkeit nachgewiesen werden [Sch01][S. 165ff.]. Die entsprechenden Applets sind über das WWW unter folgender Adresse [UM04] abrufbar.

Zur Zeit umfasst die Sammlung ca. 35 Java-Applets, hauptsächlich aus den Gebieten

Rechnernetze und Multimediatechnik.

3.1.2 Simulationen

Simulationen hingegen dienen der Überprüfung von Vermutungen, sie stellen ein möglichst authentisches Abbild von realen Vorgängen dar. Die Ergebnisse entsprechender Simulationen gehen oftmals in einen Planungs- bzw. Produktionsprozess ein, wie Chung et al. in [CHB01] darlegen.

Es gibt in der Zwischenzeit eine hohe Anzahl an Simulationen, die in der Lehre eingesetzt werden. Diese reichen von technischen Systemen, wie z. B. die Simulation von elektronischen Schaltkreisen [PAS02], bei denen ein einzelner Benutzer die entsprechenden Simulationsparameter einstellen kann, über soziale Simulationen [Sei75] bis hin zu modernen Verfahren, wie so genannte Participatory Simulations. Hierbei übernehmen die Teilnehmer Rollen und beeinflussen entsprechend die Simulationsparameter. Dadurch entsteht ein Systemverhalten (so genanntes emergent behavior), welches dann wiederum beobachtet werden kann.

Ein mögliches Anwendungsgebiet wären z. B. Börsensimulationen [WK04]. Bei diesen übernehmen die Teilnehmer die Rollen von Käufern und Verkäufern. Aus den angezeigten Kursentwicklungen können dann die Teilnehmer Erkenntnisse bezüglich der gültigen Gesetzmäßigkeiten gewinnen.

3.2 Automatisierte, isolierte Aufgabensysteme

3.2.1 Einfache Aufgabensysteme

Einfache Aufgabensysteme haben meist keine genaue Wissensmessung der Studierenden zum Gegenstand. Sie stellen, oft dynamisch, den Studierenden Aufgaben, die im Schwierigkeitsgrad ansteigend sind und nach der Eingabe die korrekte Lösung aufzeigen.

Nykänen [Nyk00] entwickelte ein System für mathematisch orientierte Aufgaben, hierbei dient das Mathematikprogramm Maple als Grundlage. Die Aufgaben werden über ein Web-Interface den Schülern präsentiert. Das Konzept ist insbesondere für Aufgaben aus dem Bereich der symbolischen Mathematik geeignet. Für nicht mathematisch orientierte Domänen ist ein Transfer allerdings nicht möglich.

Kurz und Hübner stellen ein Übungssystem aus dem Bereich der Physik vor. Hierbei wird das Wissen in Form von Multiple-Choice-Fragen abgefragt [KH02]. Das System dient hauptsächlich zum Self Assessment der Studierenden. Eine detaillierte Wissensmessung findet hierbei nicht statt.

3.2.2 Das Computerunterstützte Adaptive Testen

Sehr viele Verfahren des so genannten Computerunterstützten Adaptiven Testens (CAT) beruhen auf der Item Response Theory (IRT) von Wainer [Wai90].

Die Item Response Theory

Die relevanten Parameter nach der Item Response Theory (IRT), die sehr vielen Tests zugrunde liegt, sind folgende:

Schwierigkeitsgrad (Difficulty parameter): Dieser Faktor beschreibt den Schwierigkeitsgrad einer gegebenen Frage.

Abgrenzungsfaktor (Discrimination parameter): Dieser Parameter beschreibt, wie gut sich eine gegebene Frage auf der Skala der Schwierigkeitsgrade einordnen lässt.

Ratefaktor (Pseudo-guessing parameter): Der Faktor gibt an, wie hoch die Wahrscheinlichkeit ist, durch reines Raten die Frage richtig zu beantworten.

Die bedingte Wahrscheinlichkeit, dass die richtige Beantwortung einer entsprechenden Frage auch diese Eigenschaft tatsächlich widerspiegelt ergibt sich dann nach Hambleton & Swaminathan [HS85] in [MN99] vorgestellte 3-parameter logistic model (3PLM) wie in Formel 3.1 dargestellt.

$$P_g(\theta) = c_g + \frac{(1 - c_g) \exp[a_g(\theta - b_g)]}{1 + \exp[a_g(\theta - b_g)]} \quad (3.1)$$

Hierbei ist a_g der Abgrenzungsfaktor, b_g der Schwierigkeitsgrad und c_g der Pseudo-Rate-Faktor. θ ist der Merkmalsgrad einer gesuchten Eigenschaft.

Für die Auswahl einer Frage wurden zwei bekannte Verfahren verwendet, der maximale Informationsansatz nach Lord [Lor77] in [MN99] und die Bayessische Fragen Auswahl nach Owen [Owe75] in [MN99]

Der CBAT-2-Algorithmus

Ein weiteres Verfahren, welches die o. g. Parameter berücksichtigt, ist der so genannte CBAT-2-Algorithmus von Huang [She96]. Der Lösungsansatz um bei Multiple-Choice-Fragen eine Adaptivität zu erreichen besteht bei diesem Verfahren darin, dass zunächst eine große Anzahl an Multiple-Choice-Fragen generiert wird. Jeder Frage wird ein Schwierigkeitsgrad, sowie der entsprechende Ratefaktor zugeordnet. Diese Daten werden in einer Datenbank hinterlegt.

Die Auswahl der einzelnen Fragen erfolgt anschließend in zwei Schritten: Zunächst wird zufällig der Themenbereich ausgewählt. Dann wird eine konkrete Frage aus dem Themengebiet ausgewählt. Dies geschieht auf Basis eines gewünschten Schwierigkeitsgrades. Dieses Verfahren setzt eine große Datenbank mit erstellten Fragen in unterschiedlichen Schwierigkeitsgraden und unterschiedlichen Varianten voraus.

Trotz dieser Verfahren gestaltet sich die Konzeption entsprechender Tests als sehr aufwendig.

Eine entsprechende Übersicht über adaptives Testen, findet sich in [MN99].

3.3 Intelligente Tutorielle Systeme

Intelligente Tutorielle Systeme (ITS) sollen den Studierenden eine individuelle Betreuung zuteil werden lassen. Ergebnisse aus der Psychologie belegen, dass Studierende, die betreut werden, in Wissenstests einen um zwei Sigma verbesserten Wert aufweisen [Blo84]. Unter Verwendung von künstlicher Intelligenz sollten Studierende in einer hoch interaktiven Lernumgebung dazu motiviert werden, komplexe Aufgabenstellungen zu lösen und dadurch ihren Wissensstand und ihre Fähigkeiten zu verbessern.

3.3.1 Aufbau eines ITS

In aller Regel besteht ein ITS aus vier Komponenten, die miteinander den Ablauf einer ITS-Sitzung steuern:

Expertenmodul Das Expertenmodul modelliert einen Experten des betreffenden Wissensgebietes. Aufbauend auf dieser Wissensbasis präsentiert es Informationen, Aufgaben und gibt Lösungswege an. Die Repräsentation des Wissens erfolgt durch die einstufige Prädikatenlogik. Mit Hilfe von Fakten und Regeln

werden Objekte zueinander in Relation gesetzt. Die Programmierung erfolgt meist mit Hilfe der KI-Sprache Prolog.

Studentenmodul Das Studentenmodul bildet ein Modell des Studierenden ab. Es erfasst die Vorkenntnisse, beobachtet das Lösungsverhalten und hält die einzelnen Lösungsschritte fest.

Unterrichtsmodul Das Unterrichtsmodul gibt die Lehrinhalte vor, die zum jeweiligen Studentenmodell passen, und bestimmt den geeigneten Interaktionsstil in Abhängigkeit von den Fähigkeiten des Kommunikationsmoduls.

Kommunikationsmodul Dieses Modul ist die Schnittstelle zwischen dem ITS und dem Benutzer. Es wählt eine passende Form, wie der Interaktionsinhalt, den das Unterrichtsmodul bestimmt, präsentiert werden soll. Die Grenzen des Moduls werden sowohl durch die verwendete Hardware wie auch die Systemsoftware bestimmt.

3.3.2 Ablauf einer ITS-Sitzung

Zunächst erfasst das Studentenmodul mit Hilfe des Kommunikationsmoduls den Kenntnisstand des Studierenden. Das Unterrichtsmodul konzipiert nun zu diesem laufenden Studentenmodul einen Unterrichtsplan. Dieser wird mit Hilfe des Expertenmoduls generiert. Entsprechend den Interaktionen des Studierenden werden nun bei passenden Interventionszeitpunkten (insbesondere bei Fehlern) Rückmeldungen gegeben, um das Lernen positiv zu beeinflussen.

3.3.3 Beispielanwendungen

ITS wurden in verschiedenen Anwendungen implementiert. Eine Übersicht findet sich in [BSH01] [S. 225-232]. Insbesondere das Militär setzte ITS ein, um Offiziere und Techniker zu schulen. Zwei Systeme sollen in diesem Zusammenhang erwähnt werden: SHERLOCK schult Air-Force-Personal bei der Fehlersuche an F-15 Düsenjets und TAO (Tactical Actions Officer) schult Navy Offiziere, die Erfassungs- und Waffensysteme an Bord eines Kreuzers zu bedienen.[Lus92]

3.3.4 Ein Autorensystem für ITS: Eon

Ein sehr komplexes System, welches die Erstellung einer gesamten Lernumgebung in Form eines ITS zum Ziel hat, ist Eon [Mur98] [S. 5-64]. Es handelt sich um ein Projekt von Tom Murray. Das Projekt wurde bereits 1997 beendet, es umfasst die Gestaltung einer Lernumgebung, einer Wissensbasis, der Lehrstrategie und eines Studentenmodells. Das Projekt wurde in der Programmiersprache SK8 verfasst. Das Ziel von Eon wurde wie folgt beschrieben: "The goal of Eon is to allow instructional designers to cost-effectively build multimedia-rich cross-platform tutorials and learning environments with embedded intelligent instructional strategies."¹. Eon beinhaltet folgende Tools:

- Document Browser
Dieses Werkzeug ermöglicht die hierarchische Darstellung aller Objekte des Tutorials.
- Topic Network Editor
Er ermöglicht die grafische Darstellung der pädagogischen Beziehungen zwischen den einzelnen Themen.
- Tutoring Strategy Editor
Dieser Editor nutzt die Wissensrepräsentation in Form eines Netzwerkes und erlaubt die Gestaltung und Behandlung eines Themas, wenn er im Netzwerk aktiviert wird.
- Interaction Editor
Der Interaction Editor erlaubt die Gestaltung einer Repräsentation, die festlegt, wie die Informationen den Studierenden dargestellt werden sollen.
- Topic Contents Browser
Den einzelnen Themen können hier, wie bei Objekten, einzelne Eigenschaften zugewiesen werden. Dies betrifft z. B. die Wichtigkeit oder den Schwierigkeitsgrad.
- Presentation Contents Browser
Dieser Browser erlaubt die Gestaltung einzelner interaktiver Templates. Durch dieses Werkzeug kann unter anderem die Hintergrundfarbe gewechselt oder bestimmte Bilder eingestellt werden.

¹<http://helios.hampshire.edu/~tjmCCS/eon.www/eon.html>

3 Existierende Vorarbeiten

- Student Model Editor

Dieser Editor setzt das festgestellte Verhalten der Studierenden in Beziehung zum Thema und zur Unterrichtsgestaltung.

- Meta Strategy Editor

Durch dieses Werkzeug kann eine Veränderung der grundlegenden Lehrstrategie festgelegt werden, so zum Beispiel von der Didaktik (Theorie des Unterrichts) zur Sokratik (wertorientierte Reflexion).

Insbesondere durch den Topic Network Editor von Eon, welcher die Netzwerkstruktur zwischen den einzelnen pädagogischen Zusammenhängen darstellt, hebt sich das System hervor.

3.3.5 Kritik an ITS

Die ursprünglichen Ziele der Künstlichen Intelligenz (KI), wie die Nachbildung der menschlichen Intelligenz, konnten nicht erfüllt werden. Insbesondere sind für die Entwicklung geeigneter ITS sehr hohe Kosten und lange Entwicklungszeiten einzuplanen, da die Abbildung kompletter Wissensdomänen sehr komplex ist.

Daneben werden in der Forschung bisher nur wenig praktische Aspekte wie Software Engineering, insbesondere die Robustheit der Systeme berücksichtigt. Zudem sind einige Probleme der KI, wie z. B. die Wissensrepräsentation komplexer Inhalte, Probleme in der Darstellung hierarchischer Klassifikationssysteme grundsätzlich und damit auch für ITS bis heute nicht zufriedenstellend gelöst. Patrick Suppes führt anhand der beiden Systeme VALID und EXCHECK aus, dass man 1990 nicht weiter sei als vor 20 Jahren [Sup90]. Es bleibt festzuhalten, dass sich ITS innerhalb der langen Zeitspanne von über 30 Jahren nicht haben durchsetzen können, dass sie außerdem bis heute kein Gruppenlernen unterstützen und auch keinen Einfluss auf Kommunikationsprozesse haben. Als positives Ergebnis bleibt eine verbesserte Benutzerschnittstelle, die z. B. auch die natürlich-sprachliche Interaktion erlaubt.

3.4 Kognitive Tutorielle-Systeme

Kognitive Tutorielle-Systeme werden nach Anderson et al. wie folgt definiert: "Kognitive Tutoren sind intelligente Tutorsysteme, deren Design auf der kognitiv psychologischen Theorie basieren und welche aus Methoden bestehen, die komplexe

und authentische Probleme für Studenten präsentieren“ [ACKP95].

Es wurde mehrfach versucht, aufbauend auf einer Standard-Architektur, Systeme zu entwickeln, die es ermöglichen sollen beliebige Tutor-Systeme zusammenzustellen [Bru04], [HKA03], [CWG⁺02]. Für sehr einfache, statische Sachverhalte liefern die Systeme zufriedenstellende Ergebnisse. Komplexe Sachverhalte (insbesondere mit einer dynamischen Aufgabenstellung) müssen jedoch von Hand programmiert werden. Ein System zur Erstellung entsprechender Aufgaben wird in Abschnitt 3.7.1 noch kurz vorgestellt. Es handelt sich hierbei um das Cognitive Tutor Authoring Tool (CTAT) von Ken Koedinger.

3.5 Systeme zur Unterstützung der sozialen Interaktivität

Dowling zeigt in [Dow02], dass durch pädagogische Agenten eine soziale Interaktion mit Studierenden bewirkt werden kann und dadurch Ansätze, die ursprünglich mit Intelligenten Tutoriellen Systemen verfolgt wurden, wesentlich verbessert werden können. Sie adaptieren den Zustand der Lernumgebung, ermöglichen die Interaktion zwischen den Studierenden, da sie den gleichen Lernraum verwenden. Damit allerdings das Ziel einer Wissenskonstruktion (im Sinne des Konstruktivismus) erreicht werden kann, ist es bei solchen Systemen notwendig, den Lernenden mehr Metainformationen (s. Abschnitt 2.2.1) zur Verfügung zu stellen.

3.5.1 Talking Heads

Neben den automatisierten pädagogischen Agenten zeigen Graesser et al. die Möglichkeit, Fragen an den Studierenden durch animierte Avatare in Form von Talking Heads zu stellen [GXP01]. Hierbei wird die Agenten-Technologie verwendet und der Talking Head, AutoTutor genannt, vermittelt sowohl die Inhalte an die Studierenden wie auch die jeweilige Fragestellung.

3.5.2 Systeme zur Beurteilung der Lernleistung von Studierenden

Eine wesentliche Voraussetzung für erfolgreiche Tutoren-Systeme ist die Beurteilung der Studierenden nach ihren Fachkenntnissen und ihrer Leistungsfähigkeit. Hierbei

gibt es verschiedene grundlegende Verfahren, wie diese Beurteilung durchgeführt werden kann. Sie reichen über Selbstbeurteilungssysteme, automatisierte Systeme bis hin zu Fremdbeurteilungssystemen, bei denen andere Personen die Beurteilung durchführen.

Das Mathematical Science Inventory als System zur Selbstbeurteilung

LeBold et al. zeigen in [LBW98] die Möglichkeiten ihres Mathematical Science Inventory (MSI)-Systems auf. Das System führt eine Beurteilung der Studierenden aufgrund einer Selbsteinschätzung durch.

Hierbei dient das System der frühzeitigen Identifikation von kritischen (potentiell leistungsschwächeren), aber auch von talentierten Studierenden.

Das System übermittelt hierzu einen Fragebogen mit 150 Einzelfragen an die Studierenden. Nach Beantwortung dieser Fragen wird die automatisierte Beurteilung durchgeführt. Die positive Auswirkung dieses Systems auf den Lernerfolg konnte nachgewiesen werden.

Virtuelles Rennen als Abschlussarbeit

Eine Art der Beurteilung in spielerischer Form stellen Matko et al. in [MBB01] vor. Die Studierenden sollen im Bereich des Computer Aided Control Systems Design (CACSD) ausgebildet werden. Die Studierenden haben zunächst ein entsprechendes Design zu entwerfen, mit welchem sie eine auf Simulink/Matlab-basierende simulierte Surfregatta zu bestreiten haben. In Abhängigkeit der Ergebnisse, die in einem Wettbewerb ermittelt werden, werden die Noten für die Studierenden vergeben. Bei dieser Art eines Bewertungssystems wird davon ausgegangen, dass die besseren Kenntnisse über die physikalischen Gesetzmäßigkeiten und deren Abbildung im Regelungssystem sich in den besseren Ergebnissen dieser virtuellen Regatta niederschlagen. Einziges Kriterium in der Leistungsbewertung ist die Zeit. Das System wurde von den Studierenden begeistert als Abschlussprüfung akzeptiert.

Weitere Arbeiten, die insbesondere auch Computerspiele im Sinne von Lernsystemen zum Gegenstand haben, sind in [GAD02] vorgestellt.

Wissensaustausch durch Simulation und Diskussion

Liu et al. [LBJHH02] stellen ein System vor, bei denen Studierende Wertpapierportfolios bewerten. In der Studie werden Methoden vorgestellt, die eine webgestützte

Bewertung fördern. Hierbei werden Werkzeuge zur Wissensgewinnung und Data Mining verwendet.

Durch dieses System soll es insbesondere den Studierenden ermöglicht werden, unterschiedliche Standpunkte einzunehmen und durch einen Austausch mit anderen Studierenden den Sachverhalt besser zu verstehen. Die Studierenden können dann durch den Wert des Portfolios bewertet werden.

Das Projekt nutzt hierbei eine Simulationsumgebung als “conceptual framework”.

Anonymisiertes Peer-Assessment

Lin et al. stellen in [LLY01] ein System vor, welches als Besonderheit anonymisiert ein Peer-Assessment von Studierenden untereinander erlaubt.

In einer Evaluierung stellten sie fest, dass das System sowohl von den Studierenden akzeptiert wurde, welche gute Beurteilungen erhielten, wie auch von Studierenden mit schlechten Beurteilungen, es gab eine breite Akzeptanz.

Die Autoren weisen allerdings explizit darauf hin, dass die Betreuer des Systems positiv auf die Stimmungslage der Studierenden Einfluss nehmen sollten. Das System wurde in der Informatik eingesetzt.

Messung der Lernleistung in der ingenieurwissenschaftlichen Ausbildung

Dillon et al. [DKEV00] stellen in ihrer Arbeit ein Konzept, basierend auf der klassischen Feedback-Theorie und auf der General Systems Performance Theory (GSPT), vor. Das System wird in dem Bereich der ingenieurwissenschaftlichen Ausbildung eingesetzt.

Die Idee beruht auf einem generischen, insbesondere für menschliche Messwerte nutzbaren Konzept der Kapselung von Performance-Indikatoren. Hierbei werden die Messungen benutzt, um die oftmals vorhandenen “open loop” Strukturen zu schließen. Ein Zusatznutzen der gewonnenen Rückmeldung besteht in der kontinuierlichen Verbesserung der Vorlesung selbst. Die Durchführung der Leistungsmessung geschieht dadurch, dass eine zufällig ausgewählte Anzahl von vorher fest vorgegebenen (statischen) Testfällen den Lernern präsentiert wird. Es wird in der Arbeit darauf hingewiesen, dass diese Testfälle möglichst von Außenstehenden erstellt werden sollten. Das System wurde allerdings noch nicht vollständig realisiert, daher konnte auch keine detaillierte Auswertung durchgeführt werden.

3.5.3 JaTeK als Lernplattform für interaktives Lernen

Bei dem Java Based Teleteaching Kit (JaTeK) [SF98] handelt es sich um ein System, welches aus dem Projekt "Teleteaching Dresden-Freiburg" hervorgegangen ist. Das System hat seinen Schwerpunkt im Bereich der Kommunikation. Es unterstützt hierbei sowohl synchrone wie asynchrone Verfahren. Insbesondere werden hierbei bekannte Kommunikationsverfahren wie Chat oder Newsgroups in einer einheitlichen Oberfläche abgebildet. Zu den Kommunikationskanälen wird ein entsprechender Bezug zum Kursmaterial hergestellt. Das System steht nun als Public Domain Software unter [Sou03] der Allgemeinheit zur Verfügung. Folgende JaTeK-Komponenten werden angeboten:

- WebCall 1.0
WebCall 1.0 ist eine Software, die Webtechnologie verwendet und zusätzliche Dienste wie Annotationen, Feedback und ein entsprechendes Logging anbietet. Der Schwerpunkt liegt im synchronen Betrieb, jedoch besteht durch die Aufzeichnung der Präsentation auch die Möglichkeit der Unterstützung des asynchronen Betriebes.
- Java Based Teleteaching Kit (JaTek)
Das Modul dient im Wesentlichen der Erstellung und der Verwaltung von Kursmaterialien. Der Schwerpunkt liegt hier im asynchronen Lernmodus. Es ist auch möglich, Übungen und Tests zu erstellen.
- Java Based Workgroup Support (JaWoS)
Dieses Modul unterstützt die Gruppenkommunikation im Sinne von Computer Supported Cooperative Work (CSCW). Es werden sowohl synchrone (z. B. Chat) wie auch asynchrone Kommunikationsmöglichkeiten (z. B. Shared Document) unterstützt.
- Classroom
Die Funktionen sind dieselben wie im WebCall-Modul, nur dass hierbei noch durch eine zusätzliche Rechtevergabe verschiedene Szenarien realisiert werden können (z. B. Vorlesungen, Gruppenarbeit).

Es werden in diesem System alle zur Zeit gebräuchlichen Kommunikationsformen abgebildet. Allerdings verwendet auch dieses System für Übungsaufgaben einfache

Multiple-Choice-Tests und Lückentexte, und die Kommunikation wird auch nicht durch das System gesteuert.

3.6 Systeme zur Unterstützung von Lerngruppen

Dieser Abschnitt beschäftigt sich mit Systemen, die das Lernen in Gruppen im Sinne des sozialen Konstruktivismus (s. Abschnitt 2.5.1) unterstützen. Hierbei reichen die vorgestellten Systeme von Review-Systemen bis hin zu komplexen Lernumgebungen.

3.6.1 Praktomat

Das System Praktomat wurde an der Universität Passau von Andreas Zeller entwickelt [Zel00]. Es handelt sich hierbei um ein Übungssystem, welches im Programmierpraktikum eingesetzt wird. Den Studierenden wird zunächst eine Programmieraufgabe präsentiert, die durch einen Variantengenerator jeweils modifiziert werden kann [EGFvG03][S. 139], dadurch können bis zu 256 Varianten eines Programmierproblems erstellt werden. Nach der Bearbeitung der Programmieraufgaben können die Studierenden ihre Einsendungen dem System übergeben. Es erfolgt zunächst eine automatisierte Überprüfung auf Korrektheit (sowohl syntaktisch wie auch durch Prüfung der Ergebnisse). Das System überprüft ebenfalls, inwieweit formelle Programmierkriterien eingehalten wurden (Variablenbenennung, Mindestmaß an Kommentaren). Hierbei verwendet das System eine heuristische Prüfmethode, in der z. B. die Anzahl der Kommentierungsworte und deren Existenz geprüft wird.

Eine der Hauptaufgaben des Systems ist es jedoch, dass auch die Studierenden den Quellcode ihrer Kommilitonen einsehen und entsprechende Verbesserungsideen im Rahmen eines Review-Prozesses übermitteln können. Eine einfache Form der Gruppenarbeit (mit zwei fest definierten Rollen) wird bei diesem System somit eingeschränkt unterstützt. Allerdings findet hier keine Online-Kommunikation statt, und das System ist nicht adaptiv zum Wissensstand der Studierenden.

3.6.2 VITAL

Das System VITAL (Virtual Teaching and Learning) [PWBWYM98] stellt als kooperative Lernumgebung den Lernern verschiedene Bereiche zur Verfügung. Der Schwer-

punkt liegt neben der gemeinsamen Bearbeitung von Lernmaterialien, welche per Hypermedia miteinander verknüpft sind, auch in der Kommunikation. Hierzu stellt die Umgebung neben Chat und Audio auch eine Awareness-Funktion zur Verfügung. Die Lernumgebung bietet zudem noch virtuelle private Lernräume wie auch Bereiche, die für die Gruppenarbeit gedacht sind.

3.6.3 CROCODILE

Aufbauend auf VITAL wurde die Lernumgebung CROCODILE (Creative Open Cooperative Distributed Learning Environment) [WP01] entwickelt. Hierbei wurde das ursprüngliche System, welches nur virtuelle Räume kannte, um virtuelle Gebäude erweitert. Diese unterscheiden sich von den virtuellen Räumen dadurch, dass sie über unterschiedliche Funktionalitäten verfügen. Durch dieses System soll ein ganzer Campus abgebildet werden.

3.6.4 ExpertFinder

Ein System zum Auffinden von Mitlernern wurde in [BRW03] beschrieben. Bei dem System ExpertFinder werden sowohl statische Informationen der Benutzer wie auch deren Lernhistorie dazu verwendet, geeignete Mitlerner zu identifizieren. Das System integriert jedoch weder adaptive Übungen, noch werden Gruppenaufgaben unterstützt. Das System geht davon aus, dass ähnliche Lerner besser als Gruppe harmonisieren.

3.7 Autorensysteme zur Erstellung von Übungsaufgaben

Ein Problem bei allen Übungssystemen ist die effiziente Erstellung von Übungsaufgaben bzw. Tutorien. Aus diesem Grund gibt es bei fast allen Lernplattformen integrierte Autorensysteme zur Erstellung entsprechender Aufgaben. Meist beschränken diese sich jedoch auf einfache, statische Multiple-Choice-, Lückentext-, Sortier-, einfache statische Rechen- und Zuordnungsaufgaben.

3.7.1 Cognitive Tutor Authoring Tools

Bei den Cognitive Tutor Authoring Tools (CTAT) von Ken Koedinger [KAH03] handelt es sich um eine javabasierte Entwicklungsumgebung, die es Dozierenden erlauben soll, ohne eine lange Vorbereitungszeit zeiteffizient Übungsaufgaben zu erstellen.

Hierbei werden Aufgaben als eine final-state-machine abgebildet. Dies bedeutet, dass die einzelnen Bearbeitungsschritte durch klar definierte Zustände festgelegt werden. Hierbei werden insbesondere idealtypische Lösungsschritte und Fehler modelliert. Das Besondere an diesem System ist, dass dem Lerner je nachdem welchen Bearbeitungsschritt er durchläuft, ein angepasster Hinweis gegeben werden. Dies ist insbesondere für den Fall pädagogisch sinnvoll, wenn der Lerner einen Fehler verursacht.

Allerdings unterstützt das System keine dynamischen Aufgaben. Die Repräsentation der Aufgaben ist auf die von den Authoring-Tools vorgegebenen Schemata beschränkt.

3.7.2 Der WIL-MA-Editor

Im Rahmen des an der Universität Mannheim initiierten Projektes WIL-MA (Wireless Interactive Learning - Mannheim) [SME⁺03] wurden mehrere Editoren entwickelt, die die Dozierenden bei der Erstellung von Aufgaben unterstützen. Zunächst besteht die Möglichkeit, einfache Multiple-Choice-Fragen zu generieren, es ist aber auch möglich, Grafiken und Bilder so einzubinden, dass auf diesen Bildern interagiert werden kann. So ist es z. B. möglich, Landkarten einzulesen und die Frage nach der Lage eines bestimmten Orts zu stellen, der dann vom Lerner mit der Maus oder dem Stift angeklickt wird.

3.7.3 Unterstützung von Übungsaufgaben durch IMS QTIS

Da die Erstellung von Übungsaufgaben, selbst bei Multiple-Choice-Aufgaben, ein sehr aufwändiges Verfahren ist, ist es interessant, eine Austauschnorm zu entwickeln, die es ermöglicht, die einmal erstellten Aufgaben in andere Lernplattformen einzupflegen. Hierzu entwickelte das Instructional Management Standards Global Learning Consortium, Inc. (IMS/GLC) eine spezielle XML-Austauschnorm, die IMS Question & Test Interoperability Specification [IGLC04]. Sie enthält u. a. für folgende

Aufgabentypen genaue Spezifikationen: Multiple-Choice-Fragen, Lückentext, starre Texteingaben, Sortieraufgaben, stufenlose Rückmeldung (Slider), Assoziativ-Aufgaben und auch graphische Assoziationen. Durch diese Normen soll es ermöglicht werden, in Kombination mit den jeweiligen Test-Systemen die Aufgaben auch in anderen Lernplattformen wiederzuverwenden.

3.7.4 Zusammenfassung

Es existiert mittlerweile eine ganze Reihe von Editoren, die es den Dozierenden ermöglichen sollen, Aufgaben zu generieren. Diese reichen von sehr leicht zu bedienenden Multiple-Choice-Editoren über interaktive Bilder, gehen dann über Finite-State-Maschinen, die eine mittlere Komplexität in der Bedienung haben, bis hin zu integrierten grafischen Entwicklungsumgebungen. Diese mächtigen Werkzeuge erfordern allerdings auch eine fachgerechte Bedienung, die umfangreichere Kenntnisse voraussetzt, wie sie meist erst durch die Grundausbildung im Informatikstudium vermittelt werden. Abbildung 3.1 stellt diesen Sachverhalt noch einmal grafisch dar:

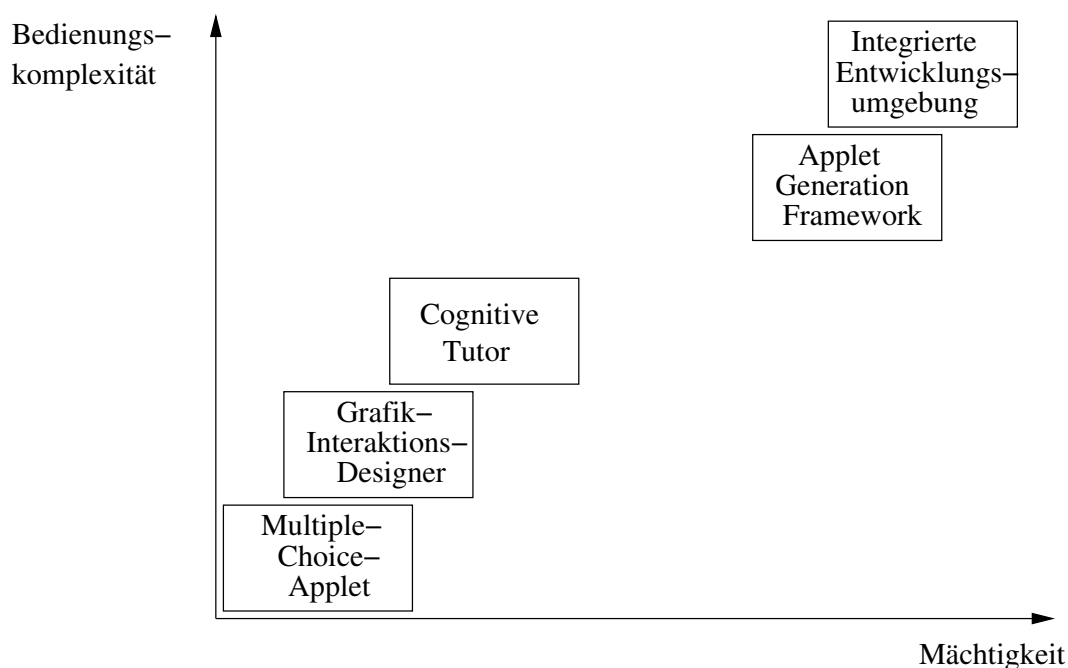


Abbildung 3.1: Einordnung von Aufgabeneditoren

Die meisten Multiple-Choice-Editoren sind zwar einfach zu bedienen, die Möglich-

keiten, Aufgaben zu präsentieren, sind jedoch auf wenige Typen (z. B. reines Auswählen von vorgegebenen Lösungsmöglichkeiten) beschränkt. Mächtige Entwicklungsumgebungen ermöglichen zwar durch das freie Programmieren das Erstellen von beliebigen Aufgabentypen, sind aber in der Bedienung oft so komplex, dass die Einarbeitungszeit, selbst bei Informatikern, meist mehrere Wochen dauert. Es ist demnach immer eine Abwägung vorzunehmen, ob der jeweilige Aufwand in einem adäquaten Verhältnis zum Nutzen steht. Eine zufriedenstellende Reduktion der Aufgabenkomplexität auf ein mittleres Niveau, wie sie etwa auch bei Präsentationsprogrammen (für den Bereich der Animationen) stattfindet und damit auch Nichtinformatikern ermöglicht, über statische Aufgaben (Multiple-Choice-Aufgabe, Lückentextaufgaben etc.) hinausgehende interessantere Aufgabentypen zu erstellen, ist eine Herausforderung. Hierzu wäre eine Authorware zu entwickeln, die den Ersteller von Übungsaufgaben durch geläufige Bedienparadigmen unterstützt.

3.8 Integrierte Lern- und Übungsumgebungen

Lernumgebungen bieten eine Infrastruktur an, durch welche Lerninhalte den Lernenden zur Verfügung gestellt werden. Sie ermöglichen neben administrativen Aufgaben auch die didaktische Unterstützung. Mittlerweile werden bei den meisten Lernumgebungen auch so genannte Übungsumgebungen integriert.

Da es in der Zwischenzeit eine kaum noch zu überschauende Anzahl an Lernumgebungen gibt (siehe hierzu z. B. [Sch03]), beschränken sich die folgenden Darstellungen auf bekannte, erfolgreiche und für CATS bezüglich einer möglichen Integration relevante Systeme.

Es werden sowohl kommerzielle wie auch Open-Source-Plattformen vorgestellt. Um eine weitestgehende Austauschbarkeit der Lerninhalte zwischen den Lernplattformen zu ermöglichen, wurde das SCORM-Modell entwickelt.

3.8.1 SCORM-Die Austauschnorm für Lerninhalte

Das Sharable Content Object Reference Model (SCORM) wurde von der Advanced Distributed Learning Initiative (ADL) 2000 vorgestellt [Ini05]. Es gab zu dieser Zeit bereits eine Vielzahl von Organisationen, welche ihren eigenen Standard zum Austausch von Lerninhalten benutzten. Dies hatte zur Folge, dass in aller Regel für jede Lernplattform die Lerninhalte neu erstellt werden musste. Daher war eine der we-

3 Existierende Vorarbeiten

sentlichen Zielvorgaben dieser Initiative die Schaffung einer Norm, die diese Mehrarbeit vermied. Es sollten bei der Umsetzung die so genannten R.A.I.D.-Prinzipien eingehalten werden:

Reuseability Die Kurse sollten *wiederverwendbar* sein. Sie sollten insbesondere auch durch andere Entwicklungswerkzeuge modifizierbar sein.

Accessability Die *Zugänglichkeit* der Kurse sollte sichergestellt werden. Insbesondere sollte es möglich sein, die Kurse nach bestimmten Inhalten zu durchsuchen.

Interoperability Die erstellten Kursen sollten *kompatibel* zu den bekanntesten Betriebssystemen, Web-Browsern und Hardwareplattformen sein.

Durability Die Kurseinheiten sollten *beständig* bleiben. Hierbei sollten insbesondere die Kurse zu neuen SCORM-Spezifikationen aufwärtskompatibel sein und keine signifikanten Modifikationen erfahren.

Um diese Anforderungen zu erfüllen wurde bei SCORM ein entsprechendes Architektur-Modell gewählt.

Die SCORM-Architektur

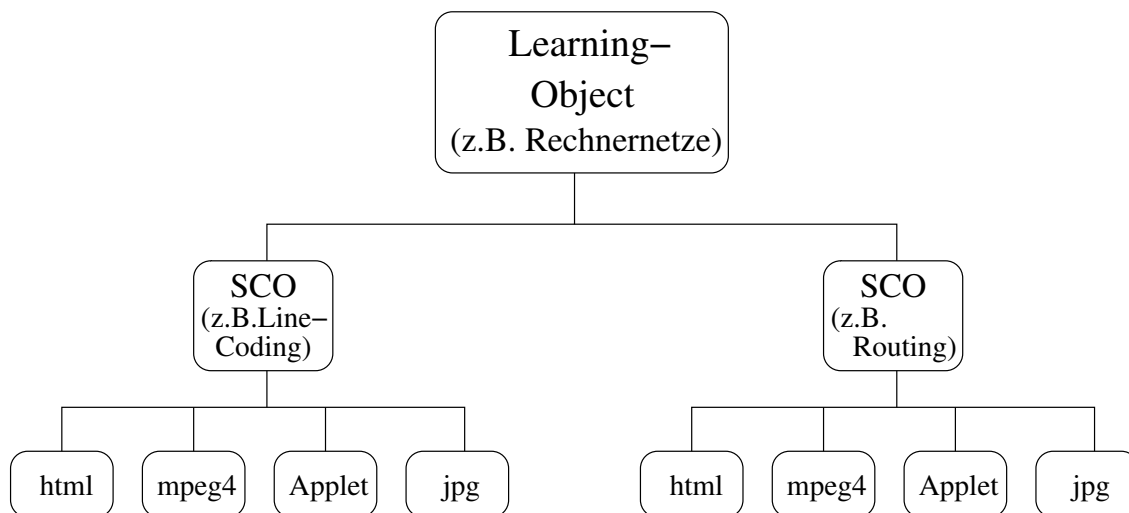


Abbildung 3.2: Die SCORM-Kurs-Architektur

Abbildung 3.2 stellt die SCORM-Architektur schematisch dar. Auf der untersten, elementaren Ebene eines SCORM-Kurses befinden sich so genannte Assets. Dies können einzelne, elementare Lern-Elemente sein, wie z. B. Webseiten, Videos, Applets oder Bilder.

Auf der zweiten Ebene befindet sich ein Sharable Content Object (SCO), es setzt sich aus einem oder mehreren Assets zusammen. Hierbei ist auf eine strikte Trennung der Inhalte von mehreren SCOs zu achten. Ein SCO darf daher niemals ein anderes SCO aufrufen oder mit ihm kommunizieren. Das Learning Management System (LMS) kommuniziert mit dem SCO über das SCORM Application Programm Interface (API).

Die nächste Ebene in der SCORM-Architektur präsentiert das so genannte Lernobjekt. Dieses besteht aus mehreren SCOs und repräsentiert einen kompletten Kurs.

Die Kurs Metadaten

Neben der inhaltlichen Gliederung der Lerninhalte, erfolgt zudem auch noch eine Beschreibung der Lerninhalte. Diese werden ebenfalls in Inhaltsstruktur-Metadaten, SCO-Metadaten und Asset-Metadaten eingeteilt. Diese Metdaten werden in einer besonderen Datei, der imsmanifest.xml-Datei festgehalten. Sie gliedert sich in vier Bereiche: Der Einleitungssektion, Metdaten-, Organisations und in die Ressourcen-sektion.

Einleitungssektion Diese Sektion enthält Informationen sowohl für das LMS als auch für den XML-Parser. Dies wird durch XML-Schemata realisiert.

Metadatensektion Diese Sektion enthält Informationen über den Kurs

Organisationssektion Dieser Bereich beschreibt den Ablauf des gesamten Kurses

Ressourcensektion An dieser Stelle werden alle Dateien aufgeführt, die von den SCOs verwendet werden.

Alle Metadaten müssen mit dem IMS Learning Resource Metadaten Model kompatibel sein. Dieses Metadatenmodell wurde aus dem Learning Object Metadaten Model (LOM) von IEEE abgeleitet.

Die SCORM-Laufzeitumgebung

Während der Laufzeit werden in dem LMS entsprechende Laufzeitmetadaten erzeugt und entsprechende Laufzeitkommandos ausgeführt. Diese werden über die SCORM-API an die SCOs des Kurses übermittelt. Die Anwendung ist in aller Regel Server-Client-basiert, wobei der Server das LMS darstellt und der Client in aller Regel ein Webbrowser ist.

3.8.2 CLIX-Campus

CLIX-Campus [icG] entstand im Rahmen des Projektes "Winfoline", in dessen Rahmen mehrere Lehrstühle aus verschiedenen deutschen Universitäten einen Masterstudiengang im Bereich der "Wirtschaftsinformatik" anbieten. Das System ist als Lernumgebung konzipiert, es wird von der Firma imc aus Saarbrücken als kommerzielles Produkt vertrieben. Das System ist mandanten- und rollenbasiert; dies bedeutet, dass durch eine Installation mehrere Institutionen verwaltet werden können und innerhalb des Systems verschiedene reale Rollen durch entsprechende Funktionalitäten abgebildet werden. Durch die Mandanten-Orientierung ist es z. B. für eine Universität möglich, einzelne organisatorische Einheiten als Mandanten zu verwalten. Diese könnten zum Beispiel von der Hochschule als Obermandant, über Fakultät, Institut bis hin zum Lehrstuhl eingerichtet werden.

Auch die umfangreiche Ausstattung an vordefinierten Rollen ist bei diesem System hervorzuheben. Diese reichen von Administratoren, gehen über Portal-Manager, diverse Redakteure, Moderatoren, Tutoren bis hin zu den Lernenden. Neben diesen vordefinierten Rollen können dem einzelnen Nutzer auch noch besondere Privilegien eingeräumt werden.

CLIX-Campus ermöglicht eine umfangreiche Verwaltung von diversen Kurseinheiten und von Belegern. Mit diesem System soll der gesamte Lehrbetrieb einer Universität abgebildet werden können. Das System wird bereits an verschiedenen deutschen Universitäten (z. B. Universität Freiburg, Universität Göttingen) eingesetzt [icG]. Das System verfügt neben den administrativen Funktionalitäten auch über Gruppenfunktionen wie z. B. Foren. Diese können kursindividuell eingerichtet werden. Zusätzlich bietet das System noch einen Übungsbereich, für den einfache Übungsaufgaben erstellt werden können. Die Art der Aufgaben ist beschränkt auf Multiple-Choice-Fragen, Lückentexte und Zuordnungsaufgaben. Als Nachrichtenservices bietet CLIX die Funktionen Chat, Forum, Schwarzes Brett an.

3.8.3 WebCT

Bei WebCT [web02] handelt es sich um eine Lernplattform, die ihre Ursprünge an der University of British Columbia hat, daher liegt auch der Schwerpunkt dieses Systems in der Hochschulausbildung. Hierbei konzentrieren sich die Funktionalitäten auf so genannte "Kurse". Diese können unterteilt und über Inhaltsverzeichnisse strukturiert werden. Die Lernplattform kennt fest vorgegebene Rollen: Administrator, Tutor, Designer und Studierende. Jeder neue Benutzer muss mindestens einer dieser Gruppen zugeordnet werden, deren Rechte er übernimmt. Kurse werden zunächst durch den Systemadministrator angelegt und können dann selbständig von der Rolle "Designer" weitergepflegt werden.

In WebCT ist es möglich, Kurse kompatibel zum Sharable Content Object Reference Model (SCORM) zu im- und exportieren. Neben den geläufigen Administrationsfunktionen unterstützt WebCT insbesondere die Kommunikation in Foren, Mail, Chat und im Rahmen eines Whiteboards. Es werden zur Unterstützung einer Evaluation die Werkzeuge Test/Umfrage, Selbsttest, Notenliste und Aufgaben angeboten [Sch03][S. 249-255].

3.8.4 Das Blackboard Learning System

Bei dem Blackboard Learning System handelt es sich um eine kommerzielle Lernumgebung. Das Unternehmen wurde 1997 gegründet und bietet sowohl für Unternehmen wie auch für Hochschulen sein Blackboard Learning System an.

Es handelt sich hierbei um eine webbasierte Client-Server-Architektur. Das System bietet neben den normalen Administrationsfunktionalitäten eine offene Java-Schnittstelle (API) an, mit der weitere Produkte in das System integriert werden können. Dies gehört zu dem Konzept des so genannten Building-Blocks-Program.

Durch ein Software Development Kit (SDK), welches alle notwendigen Java APIs und die dazugehörige Dokumentation enthält, ist es für Eigenentwicklungen möglich, auf die Funktionen der Blackboard-Plattform zuzugreifen. Dieser "Building Block" kann dann auch von jeder anderen Blackboard Installation verwendet werden.

Die gesamte Lernplattform ist SCORM 1.2 kompatibel. Nähere Informationen können über die Web-Adresse [Bla02] des Unternehmens abgerufen werden.

3.8.5 Lernraum virtuelle Universität

Der “Lernraum Virtuelle Universität” (LVU) ist die integrierte Lernumgebung der FernUniversität in Hagen. Es handelt sich hierbei um die Zusammenführung der beiden Systeme “Virtuelle Universität” und “WebAssign”. Der Lernraum virtuelle Universität ist ein Open-Source-Projekt, nähere Informationen, sowie die entsprechenden Programme sind über [dICbdiH04] abrufbar.

Die Virtuelle Universität

Das System “Virtuelle Universität” (VU) bietet verschiedene administrative Funktionalitäten wie z. B. An- und Abmeldung von Studierenden oder Inhaltsfreigabe. Daneben stellt es noch eine einfache Kursbetreuung zur Verfügung. Dies ermöglicht einerseits das Einstellen von Mitteilungen im jeweiligen Kursbereich und andererseits die Versendung von Nachrichten per E-Mail an die Studierenden. Des Weiteren stellt das System den Studierenden den Inhalt verschiedener Kurse, welche an der FernUniversität in Hagen angeboten werden, zur Verfügung. Hierzu ist es jedoch notwendig, dass sich die Studierenden zuvor in die jeweiligen Kurs eingeschrieben und diese auch bezahlt haben. Das System kennt folgende Rollen: VUAdministratoren, Bereichsadministratoren, Mitarbeiter und Studierende.

WebAssign

Die Eigenentwicklung WebAssign ist ein System für den Übungsbetrieb über das Internet [BHSV99], welches ebenfalls über die OpenSource Initiative CampusSource des Landes Nordrhein-Westfalen abrufbar ist.

Die FernUniversität in Hagen bezeichnet WebAssign selbst als “eine webbasierte Softwarekomponente, die den universitären Übungsbetrieb in das Konzept der virtuellen Universität integriert” [LPIISE01][S. 4].

Der ursprüngliche Grundgedanke dieses System ist die Abbildung des klassischen Postversandes der Übungen auf ein webbasiertes System. Hierzu werden zunächst Übungsaufgaben statisch als Dokument erstellt. Diese Übungsaufgaben werden den Studierenden zu festgelegten Zeitpunkten (in aller Regel in einem 14-tägigen-Zyklus) zur Verfügung gestellt. Diese haben die Möglichkeit, ihre Lösungen ebenfalls in elektronischer Form an die FernUniversität zurückzuschicken. WebAssign verteilt nach einem vorher festgelegten Zuordnungsschema die Lösungen an die Korrektoren, diese geben ihre Bewertung in WebAssign ein. Schließlich erhalten die Studie-

renden nach Ablauf der Einsendefrist die Musterlösung inklusive ihrer individuellen Bewertungen. WebAssign unterstützt neben statischen Dokumenten sowohl Multiple-Choice-Aufgaben, Zuordnungsaufgaben und numerische Aufgaben, die eine automatisierte Korrektur erlauben, als auch die Einbindung von Zusatz-Korrektur-Modulen, die z. B. eine automatisierte Vorkorrektur von Programmieraufgaben durch Kompilation erlauben. Dieses System ist jedoch weder adaptiv noch werden Gruppenfunktionalitäten unterstützt noch existiert eine integrierte Kommunikationsunterstützung. Jeder Studierende erhält zudem die gleiche Aufgabe.

3.8.6 Die Lernplattform .LRN


Es handelt sich bei .LRN[LR04] um ein Open-Source-Projekt (auf Basis der GNU-Lizenz) von verschiedenen Universitäten (u. a. auch Heidelberg, Bergen, MIT-Sloan und Mannheim). Das System ist SCORM-kompatibel dadurch ist es möglich, Lerninhalte, die auf anderen Systemen erstellt wurden, in diese Lernplattform einzulesen bzw. vorhandene Lerneinheiten zu exportieren. Zum Zeitpunkt der Erstellung dieser Arbeit war die .LRN-Umgebung SCORM 1.2 kompatibel. Abbildung 3.3 zeigt die personalisierte Einstiegsseite, wie sie den Studierenden der Universität Mannheim präsentiert wird. Auf der linken Seite sind die Links zu den belegten Veranstaltungen zu sehen, im zentralen Bereich der personalisierte Kalender. Weitere Informationsdienste sind am Rand zu erkennen. Da zwischen der .LRN-Umgebung und dem CATS-System eine Ergebnisübermittlung stattfinden soll, wird im Folgenden die Lernplattform detaillierter beschrieben.

Funktionalitäten des Systems

Das System bietet neben der klassischen Benutzerverwaltung und Administration als Besonderheit einen gemeinsamen Arbeitsbereich für Studierende, in denen sie beliebige Files austauschen können. Hierdurch soll das kooperative Lernen unterstützt werden, was insbesondere im Sinne des Konstruktivismus sehr förderlich ist. So können die Studierenden z. B. eigene Vorlesungsaufzeichnungen dort hinterlegen, oder gemeinsame Arbeitsvorbereitungen (z. B. für Praktika) können dort bearbeitet werden. Hierzu verwendet das System das bekannte System WWW Distributed Authoring and Versioning (WebDAV). Durch dieses System wird die Nur-Lese-Eigenschaft des WWW überwunden und es wird nun editierbar.

Daneben werden auch Diskussionsforen unterstützt, die sowohl moderiert als auch

3 Existierende Vorarbeiten



UNIVERSITÄT
MANNHEIM

Hans Christian Liebig

Eigene Startseite

Hilfe

Abmelden

Eigene Startseite

Eigener Kalender

Eigene Dateien

Verwalten

Lehrstühle und Gruppen

[Lehrstühle oder Gruppen hinzufügen/entfernen]

Lehrstühle:

- Wirtschaftsinformatik III
 - OOE II - Objektorientierte Entwicklung in Java
- Produktionswirtschaft und Controlling
 - Kosten- und Erlösrechnung
- Informatik V
- Rechnerarchitektur

Gruppen:

- e-Learning Gruppe

Diskussionsforen

Kosten- und Erlösrechnung

- Kosten- und Erlösrechnung Forum

OOE II - Objektorientierte Entwicklung in Java

- OOE II - Forum

e-Learning Gruppe

- e-Learning Gruppe Forum (Fragen zu dotLRN)

Häufig gestellte Fragen (FAQs)

Name	Gruppe
Alles um die Veranstaltungen in KER	Kosten- und Erlösrechnung
e-Learning Gruppe FAQ	e-Learning Gruppe

Tagesplan

16. September 2004

07:00
08:00
09:00
10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
18:00
19:00
20:00
21:00
22:00

Weblogger

No Entries

News Aggregator

FAZ.NET, 2004-09-16 12:12:06

RSS

Abbildung 3.3: Die .LRN-Lernplattform an der Universität Mannheim

unmoderiert sein können. Eine weitere Besonderheit ist die erweiterte E-Mail-Funktionalität, so können Nachrichten gesammelt und als Block verschickt werden. Auch ist die Nachrichtenübersendung nicht auf E-Mail beschränkt, es können zum Beispiel auch Kurznachrichten (SMSe) zu Mobiltelefonen geschickt werden. Auch verfügt .LRN über eine Gruppen-Unterstützung. Es werden Teams, Communities und Untergruppen unterstützt, die jeweils ihre eigenen Foren definieren können. Als Koordinationsfunktionalität steht eine Kalenderfunktion zur Verfügung. Diese kann von den Studierenden im Rahmen der Anpassung mit ihrer jeweiligen Startseite verknüpft werden. Daneben werden noch E-Mails, News unterstützt. Zur Unterstützung von Hausübungen gibt es eine so genannte "Homework Dropbox", die ähnlich wie bei WebAssign die Distribution der Aufgabenstellung und der Lösun-

3.9 Systeme zur Unterstützung der synchronen Lerner-Kommunikation

gen bzw. Musterlösungen übernimmt. Zum Zeitpunkt der Erstellung dieser Arbeit ist ein Modul zur Unterstützung von Evaluationen und Assessments in Entwicklung.

Die .LRN-Plattform hat auch zur Unterstützung der synchronen Kommunikation zwischen den beteiligten Personen eine Chat-Funktion. Hierbei können Studierende, die zeitgleich in der Plattform angemeldet sind, sich mittels Texteingabe "unterhalten". Eine Audio-/Videounterstützung wird derzeit nicht angeboten.

Implementierung

Die Lernplattform .LRN ist in drei Schichten aufgebaut: Auf der untersten Schicht, der Datenbankschicht, läuft eine relationale Datenbank (z. B. PostgreSQL). Darauf aufbauend läuft das System OpenACS 5 (Open Architecture Community System) [Tea04], welches Funktionalitäten zur Unterstützung von Web-Communities bereitstellt. Darauf aufbauend wiederum wurden die einzelnen .LRN-Module entwickelt. Auf der Seite der Clients werden Standard-Webbrowser verwendet.

Neben den integrierten Lern- und Übungsumgebungen, die auch eine unimodale Kommunikation zwischen den Lernern unterstützen, gibt es Einzelsysteme zur Unterstützung der synchronen Lerner-Kommunikation.

3.9 Systeme zur Unterstützung der synchronen Lerner-Kommunikation

Fast alle heute verfügbaren Lernplattformen unterstützen die Kommunikation zwischen den Lehrenden und den Lernern. Meist werden asynchrone Verfahren (Gruppen, E-Mail, Foren, etc.) verwendet. In einigen modernen Lernplattformen wird auch eine synchrone Kommunikationsmöglichkeit in Form von (unimodalen) Chats angeboten. Doch diese Art der Kommunikation reicht zur Vermittlung komplizierter Wissensinhalte alleine nicht aus. Wie bereits in Abschnitt 2.2 dargestellt, existiert in vielen Modellen eine Phase, in denen ein intensiver Erfahrungsaustausch mit anderen Lernern von entscheidender Bedeutung ist. Die rein textuelle Übermittlung, reicht hierbei nicht aus. Denn es fehlt die Übermittlung von non-verbalen Reizen, der Informationsfluss wird durch die Tastatureingabe stark gebremst. Die Texteingabe ist mit einem zusätzlichen Aufwand verbunden, so dass Wissensinhalte zum

Teil stark komprimiert werden und damit die erfolgreiche Internalisierung behindern. Die Übertragung, selbst bei einem synchronen Chat ist doch mit deutlichen Verzögerungen verbunden, so dass spontane Reaktionen, die für eine gute Interaktion notwendig sind, teilweise unterbunden werden.

In der Lernwissenschaft wird insbesondere darauf hingewiesen, dass die Einbettung in ein bedeutungsvolles Umfeld das Erinnerungsvermögen steigert und die Einordnung von Diskussionen verbessert siehe hierzu [BCD89] in [CHWS⁺02]. Insbesondere zeigten [SM94] in [WM03] das Gruppen, die Text-basiert zusammenarbeiten mehr Zeit für Entscheidungen benötigen und ihre Produktivität schlechter ist. Wie bereits in Abschnitt 2.4 gezeigt, haben sehr viele Modelle der Wissensvermittlung eine Phase, in der der Austausch mit anderen über das jeweilige Wissensgebiet im Vordergrund steht. Eine effiziente Kommunikation ist allerdings hierbei nur durch eine direkte synchrone Kommunikation möglich, da nur hierdurch Interaktionen ohne nennenswerten Zeitverzug möglich sind und ein Gemeinschaftsgefühl entsteht.

3.9.1 Audio-/Video-Konferenz-Systeme

Im Bereich der Audio-/Video-Konferenz-Systeme existieren sehr viele Technologien. Zurzeit sind Systeme, die auf der Norm H.323 beruhen, sehr weit verbreitet. Diese Norm gestattet die Durchführung von Videokonferenzen sowohl über das Telefonnetz (allerdings basierend auf ISDN), als auch über IP. Eine sehr weit verbreitete Software, die diese Norm unterstützt, ist Netmeeting von der Firma Microsoft [Mic02b]. Diese Software gehört zum Lieferumfang der Windows-Betriebssysteme.

Hierzu ist jedoch zu bemerken, dass Microsoft in zukunftsorientierten Produkten (wie z. B. dem MS-Messenger) das Session Initiation Protocol (SIP) einsetzt. Insbesondere SIP wird im Moment immer stärker in neuen VoIP-Systemen eingesetzt. Daher erfolgte im Rahmen dieser Arbeit auch eine entsprechende Integration. In Abschnitt 5.3 werden die beiden Normen H.323 und SIP ausführlich dargestellt.

3.9.2 Instant-Messenger-Dienste

Die so genannten Instant-Messenger-Dienste ermöglichen es den Teilnehmern, zu einem vorher definierten Kreis von Personen eine ständige Verbindung aufzubauen. Hierzu ist es zunächst notwendig, dass sich die Teilnehmer bei den jeweiligen Diensten registrieren. Anschließend können sie sich jederzeit von beliebigen Rechnern aus anmelden. Ihre Anwesenheit wird all den Personen unmittelbar angezeigt, die zu

dem so genannten Freundeskreis gehören. Hierdurch soll eine synchrone Spontankommunikation ermöglicht werden. Diese kann in Form von Chats, dies sind textuelle Kurzmitteilungen, aber bei einigen Systemen auch durch Audio- und Videoübertragung geschehen.

ICQ

Ein sehr populärer Dienst ist der so genannte ICQ-Dienst (ein Akronym von I seek You) [Inc02]. Zunächst ist, wie bei den meisten anderen Instant-Messenger-Systemen, eine Registrierung erforderlich. Hierbei hat man die Gelegenheit, weitere Informationen über sich selbst einzugeben. Dies betrifft zum Beispiel Hobbys, Musikgeschmack, Beruf, Sprachkenntnisse etc. Dies wird bei diesem System dazu genutzt, dass man neue Personen kennenlernen kann, die über gemeinsame Interessen verfügen. Dieser Chat-Dienst wird, wie eine eigene Untersuchung belegt [LE04b], von den Studierenden in Mannheim mit Abstand am häufigsten verwendet.

MSN-Messenger

Der Messenger-Dienst der Firma Microsoft wird in der Zwischenzeit mit den Betriebssystemen (Windows-XP) der Firma Microsoft ausgeliefert. In Verbindung mit dem so genannten Passport-Service (einem Dienst zur Benutzer-Authentifizierung und Abrechnung) stehen den Nutzern weitere Mehrwertdienste zur Verfügung. Der MSN-Messenger verwendet bereits das SIP-Protokoll zur Audio- und Videotelefonie [Mic02a].

AOL-Messenger

Der Messenger von America Online (AOL)[Ame02] ist als Komponente im Netscape-Browser enthalten und ist daher sehr weit verbreitet, wobei die Funktionalität dieses Dienstes sehr eingeschränkt ist und sich zurzeit ausschließlich auf die Visualisierung der Anwesenheiten und die Übermittlung von Kurzmitteilungen beschränkt.

3.10 Zusammenfassung

Es bleibt festzuhalten, dass bisher zwar einzelne Aspekte der Anforderungen der Lernpsychologie umgesetzt wurden. Es existieren eine Vielzahl von Animationen und Simulationen, die das entdeckende Lernen unterstützen. Automatisierte, adaptive Aufgabensysteme sind in Bezug auf Multiple-Choice-Fragen sehr gut verstanden, bereiten aber dennoch aufgrund der starken Beschränkung der Lösungsmenge erhebliche Probleme bezüglich der Beurteilung des individuellen Wissens.

Die Versprechen der Intelligenten Tutoriellen Systeme konnten bis heute nicht zufriedenstellend erfüllt werden, sie sind aber trotz verbesserter Mensch-/Maschinen-Schnittstelle auf die Interaktion zwischen dem Menschen und der Maschine fixiert und unterstützen kein Gruppenlernen im Sinne des sozialen Konstruktivismus. Anstelle der ITS wurden in den letzten Jahren öfters so genannte kognitive Tutoren eingesetzt. Sie sind jedoch sehr statisch und können nur für einzelne konkrete Aufgabenstellungen angepasstes Feedback geben.

Schließlich fehlen im Bereich der Übungssysteme noch Aufgabeneditoren, die es auch Nicht-Informatikern gestattet, leistungsfähige, interessante und adaptive Aufgaben zu erstellen, die über statische Multiple-Choice-Aufgaben, Lückentexte, Zuordnungsaufgaben, etc. hinausgehen.

Lösungen im Bereich von Gruppenlernen konzentrieren sich zurzeit auf gemeinsame Arbeitsbereiche, die dem Dokumentenaustausch dienen. Vereinzelt existieren auch Systeme, die rollenbasiertes Lernen unterstützen. Hierbei sind jedoch die Rollen fest vorgegeben und eine wissensbasierte Audio-/Video-Kommunikation zwischen den Lernern wird nicht unterstützt.

Es zeichnet sich insgesamt der Trend ab, dass sowohl Aufgabensysteme, wie auch die synchrone Kommunikation zunehmend in existierenden Lernplattformen integriert werden. Diese Kommunikationsunterstützung beschränkt sich zurzeit jedoch auf Chat-Tools. Diese unimodale Kommunikation ist jedoch für einen effizienten Wissensaustausch zwischen den Lernern nicht ausreichend. Hier wäre eine Audio-/Videokonferenzunterstützung sinnvoll. Eine aus pädagogische Sicht sinnvolle rollenbasierte Gruppenarbeit wird ebenfalls noch nicht unterstützt.

4 Anforderungen an ein Kommunikations- und Tutoring System für Lerngruppen

Aus den vorgestellten erkenntnistheoretischen Grundlagen, psychologischen Modellen und existierenden Vorarbeiten lassen sich nun Anforderungen an ein Kommunikations- und Tutoring-System für Lerngruppen im Internet formulieren, welches insbesondere auch im Fernlehrszenario eingesetzt werden soll.

4.1 Unabhängigkeit von Wissensgebieten

Das System soll universell für verschiedene Fächer einsetzbar sein. Hierbei soll es insbesondere möglich sein, auf die besonderen Belange des jeweiligen Faches einzugehen. Exemplarisch seien an dieser Stelle mögliche Anforderungen unterschiedlicher Fächer dargestellt. Bei Verwendung im mathematischen Bereich sollte z. B. die korrekte Darstellung von Formelzeichen sichergestellt sein. Auch die Anwendung von Beweisverfahren ist von besonderem Interesse. Bei einem künstlerisch/musischen Fach sollte der Gestaltungspielraum erhöht sein, die Studierenden sollten im Sinne des Konstruktivismus (s. Abschnitt 2.3.3) motiviert werden, eigene Artefakte zu schaffen oder die Artefakte anderer zu bearbeiten.

4.2 Adaptivität an den Leistungsstand der Studierenden

Damit die Studierenden weder unter- noch überfordert werden, sollten sich die Aufgaben an den Wissensstand und die Leistungsfähigkeit dynamisch anpassen. Somit

wird eine aus individueller Sicht interessante Aufgabenstellung, wie sie bereits in Abschnitt 2.5.1 genannt wurde, sichergestellt.

4.3 Kommunikationsunterstützung

Insbesondere in der Fernlehre ist, wie in Abschnitt 2.5.3 gezeigt, gerade die Entfernung der einzelnen Studierenden untereinander ein sehr großes Kommunikationshemmnis. Die räumlichen und zeitlichen Divergenzen sind zu überwinden. Hierbei soll es gerade nicht zu einer Durchbrechung des Fernlehrszenarios kommen, sondern es sollen technische Möglichkeiten genutzt werden, die den Fernstudierenden sowieso zur Verfügung stehen. Hierbei müssen eine zeitgemäße PC-Ausstattung sowie eine adäquate Internet-Anbindung vorausgesetzt werden. Die Kommunikationsunterstützung soll sich nahtlos in die übrigen Komponenten des zur erstellenden Systems einfügen.

Es ist auch in diesem Punkt auf die Verwendung von Standardtechnologien zu achten, die zum einen verfügbar sind und zum anderen auch in der Zukunft weiter durch eine ausreichende Anzahl von Anbietern gepflegt werden.

4.4 Unterstützung von Gruppenarbeit

Wie in Abschnitt 2.5.1 dargelegt, ist Gruppenarbeit ein sinnvolles Instrument zur Unterstützung von Lernprozessen. Hierbei sind jedoch, wie oben ausgeführt, bestimmte Bedingungen zu erfüllen: Die Gruppenaufgabe soll ein gemeinsames Ziel enthalten, mit dem sich jedes Gruppenmitglied identifizieren kann. Das System soll durch entsprechende Rollendefinition die individuelle Verantwortlichkeit klar definieren. Die Aufgabe muss so konzipiert sein, dass sie nur als Gruppe zu bewältigen ist. Hierbei ist auf eine interessante, reizvolle Aufgabenstellung zu achten, um motivierend auf die Gruppe zu wirken. Schließlich ist auf eine Ressourceninterdependenz zu achten, die in Kombination mit der ausschließlichen Bewältigbarkeit der Aufgabe als Gruppe die Motivation zur Kommunikation erhöht.

4.5 Allgemeine Anforderungen

Neben den oben genannten besonderen Anforderungen sind auch allgemeine softwaretechnische Anforderungen einzuhalten, wie z. B. :

- **Korrektheit**
Dies bedeutet, dass das System fehlerfrei seine Aufgabe erfüllt.
- **Robustheit**
Das System muss bei jeder Form der externen Kommunikation, insbesondere auch bei falschen Eingaben, sinnvoll reagieren.
- **Sicherheit**
Das System muss das beabsichtigte Verhalten sicherstellen. Insbesondere sind Eingriffe nicht autorisierter Personen durch geeignete Schutzmaßnahmen zu vermeiden.
- **Zuverlässigkeit**
Das System ist so zu konzipieren, dass selbst beim Eintritt eines Fehlerfalles dieser nur geringe Auswirkungen hat.
- **Benutzerfreundlichkeit**
Das System soll sowohl erfahrenen als auch unerfahrenen Benutzern eine komfortable Bedienung ermöglichen. Hierbei sind wichtige Aspekte die Entwicklung einer Benutzeroberfläche nach softwareergonomischen Kriterien (insbesondere Gestaltung der Oberfläche in Fenstertechnik, adäquater und systematischer Einsatz von Farben und Formen bei der Darstellung), die ständige Verfügbarkeit von Hilfefunktionen in Abhängigkeit des Systemzustands und die Fähigkeit, die Art der Interaktion in einem gewissen Umfang an den Kenntnisstand des Benutzers anzupassen. Auch die Benutzerdokumentation ist in entsprechender Qualität zu erstellen.
- **Effizienz**
Das System ist sowohl hinsichtlich seiner Laufzeit wie auch seines Speicherbedarfs effizient zu gestalten.
- **Wartbarkeit**
Das System ist so zu gestalten, dass das Suchen und Beheben von Fehlern und

4 Anforderungen an ein Kommunikations- und Tutoring System für Lerngruppen

die Verbesserung des Systems durch die Eigenschaften selbst unterstützt werden.

- Portierbarkeit

Das System selbst soll leicht auf andere Hardware-Plattformen und Betriebssysteme übertragbar sein.

4.6 Didaktische Anforderungen

Wie in Abschnitt 2.3 dargestellt, sind verschiedene Lerntheorien in der pädagogischen Psychologie anerkannt. Diese sind im Einzelnen: Der Behaviorismus, der Kognitivismus, der Konstruktivismus und schließlich der soziale Konstruktivismus. Je nach zu erlernendem Gebiet und Gestaltung soll das System die jeweiligen Lerntheorien unterstützen. So kann es sinnvoll sein, z. B. einen Vokabeltrainer im Sinne des Behaviorismus zu konzipieren, während z. B. das Erlernen von Routing-Algorithmen besser durch Simulationsaufgaben vermittelt wird (im Sinne des Konstruktivismus). Aus diesem Grund soll das zu entwickelnde System unabhängig von einer speziellen Lerntheorie sein. Es soll möglich sein, je nach Lernziel unterschiedliche Lerntheorien zu Grunde zu legen.

4.7 Wirtschaftliche Anforderungen

Da die Entwicklung neuer Übungsaufgaben Ressourcen bindet, ist auf eine angemessene Zweck-/Mittel-Relation zu achten. Dies erfordert insbesondere die Beachtung wirtschaftlicher Grundsätze. Hierbei sind ständig die zu erwartenden Kosten, die mit zukünftigen Entwicklungsprozessen und deren flankierenden Maßnahmen verbunden sind, zu beachten. Insbesondere bedeutet dies, dass ein entsprechendes System aufgrund der Architektur ohne großen Aufwand in andere Systeme (insbesondere Lernplattformen) integrierbar sein muss. Ein weiterer Punkt ist die Notwendigkeit einer möglichst leichten Inhaltserstellung.

4.8 Bewertbarkeit der Lehr- und Lernleistung

Durch das zu erstellende System soll eine Bewertbarkeit der Lehr-/Lernleistung ermöglicht werden. Insbesondere sollen Möglichkeiten geschaffen werden, die ge-

wonnenen Daten auch über mehrere Vorlesungszeiträume zu speichern und auszuwerten. Dadurch sollen Abweichungen im Zeitablauf darstellbar sein. Eine genaue Zuordnung der zu erwartenden Leistung soll ermöglicht werden. Die Studierenden sollen auf die Abschlussprüfung durch das System adäquat vorbereitet werden. Hierbei sollen die Aufgaben pro Lern-Item mindestens das Klausurniveau erreichen können.

Es lassen sich folgende Kritikpunkte bei bisherigen Bewertungsverfahren der klassischen Lehre feststellen:

- Aufgrund fehlender einheitlicher Methodik hohe Willkürgefahr
Klausuren zu demselben Thema schwanken im Schwierigkeitsgrad. Dadurch entsteht insofern eine zufällige Selektion, als dass Studierende mit einer leichten Klausur "Glück" haben können bzw. mit einer schwereren Klausur "Pech". Dies sollte durch ein einheitliches, durchgängiges Schwierigkeitsniveau vermieden werden.
- Prüfungen erfüllen nicht alle wissenschaftlichen Ziele
Heutige universitäre Prüfungen indizieren kaum wissenschaftliche Leistungsfähigkeit, da sowohl die Innovationsfähigkeit wie auch die Auseinandersetzung mit der aktuellen wissenschaftlichen Literatur kaum abgeprüft wird.
- Gruppenarbeit wird nur individuell bewertet
Die Gruppenarbeit als solche wird nicht bewertet, in den meisten Prüfungsordnungen muss der individuelle Beitrag bewertet werden, was jedoch einem wesentlichen Motivator der Gruppenarbeit (s. Abschnitt 2.5.1) entgegenwirkt.
- Bewertungsmaßstäbe sind oft nicht transparent
Bewertungsmaßstäbe sind in einigen Fächern, in denen subjektive Einschätzungen dargelegt werden müssen (z. B. in gesellschaftswissenschaftlichen Fächern), nicht transparent. Dadurch wird eine Korrektur der Lernprozesse bei den Studierenden erschwert.
- Schwierige Nachvollziehbarkeit
Aus den oft nicht transparenten Bewertungsmaßstäben ergibt sich ein Problem in der Nachvollziehbarkeit der Leistungsbeurteilung.

Die oben genannten Nachteile im bisherigen Leistungsbeurteilungssystem führen insbesondere zu folgenden Effekten:

4 Anforderungen an ein Kommunikations- und Tutoring System für Lerngruppen

- Demotivierung des Einzelnen bei negativen Erfahrungen
Die Studierenden werden durch die harte Selektionswirkung von Noten ausschließlich extrinsisch, insbesondere durch eine drohende Bestrafung (Verlust des Prüfungsanspruchs, reduzierte Berufsaussichten aufgrund schlechter Noten) motiviert.
- Ungerechtigkeitsempfinden
Den für die Studierende oft nicht nachvollziehbaren Bewertungsmaßstäbe stellen sie ihre eigene Leistungsmaßstäbe gegenüber, die sie in aller Regel als erfüllt ansehen. Testierte Fehlleistungen werden dann als "ungerechte Behandlung" empfunden.
- Entwicklung von persönlichen Antipathien
Da die Bewertung der Einzelleistungen und die darauf aufbauende Notengebung immer von Menschen durchgeführt werden, entwickeln sich bei negativer Bewertung persönliche Antipathien. Dies ist wiederum ungünstig für den Lernprozess, der ja gerade einen offenen Kommunikationsprozess voraussetzt.
- Fehlende Verbesserungsmöglichkeit nach einer Prüfung
Eine einmal durchgeführte Lerneinheit wird in der Regel durch eine unmittelbar anschließende Prüfung endgültig abgeschlossen. Ein vertiefender, anschließender Lernprozess wird nicht unterstützt. Dies wäre aber gerade im Hinblick auf eine spätere berufliche Tätigkeit sinnvoll, da innerhalb des Studiums Inhalte einzelner Fächer nur angerissen werden können und die berufliche Tätigkeit darüber hinaus weitere Kenntnisse voraussetzt. In neueren Prüfungsordnungen werden deshalb zunehmend "Freischussregelungen" aufgenommen, die unter bestimmten Voraussetzungen ein Wiederholen von bestandenen Prüfungen ermöglichen.
- Konzentration auf (nur) klausurrelevante Inhalte
In diesem Zusammenhang steht auch die Reduktion auf das "Wesentliche", welches sich auf typische, einige wenige klausurrelevante Themen konzentriert. Eine freiwillige Vertiefung, die nicht auch zur gesamten Notengebung beiträgt, fällt in den allermeisten Fächern der studentischen Optimierung zum Opfer.
- Entwicklung hin zu Einzellernern, fehlende Erfüllung der gesamtgesellschaftlichen Aufgabe

Da immer nur die individuelle Leistung in die Gesamtbenotung eingeht und wegen der harten Selektionswirkung von Noten auch eine Konkurrenzsituation vorhanden ist, erfolgt eine Erziehung hin zu Einzellernern. In Übungen ist hierzu immer wieder zu beobachten, dass trotz intensiver Einwirkung seitens der Tutoren keine Lerngruppenbildung stattfindet.

4.9 Allgemeine Verbesserungsmöglichkeiten

Die oben genannten Punkte beeinflussen den Lernerfolg negativ. Sie sind daher insoweit durch ein neues System zu verbessern.

4.9.1 Verbesserungsansätze des Beurteilungssystems

Ein Hauptpunkt ist das Beurteilungssystem selbst, es legt letzten Endes die Noten fest und ist daher einer der Hauptmotivatoren.

- Zielvorgabe durch die Dozierenden und neutrale Auswertung
Zunächst sind genaue Lernziele durch die Dozierenden festzulegen. Hierzu sind diese nicht nur qualitativ, sondern soweit wie möglich auch quantitativ zu beschreiben. Ein Beispiel einer qualitativen und quantitativen Beschreibung: "Die Studierenden sollen die Prinzipien von Differentialgleichungssystemen verstanden haben. Sie sollen in der Lage sein, lineare Differentialgleichungssysteme bis zum dritten Grad und zwei Variablen zu lösen."
- Durchgängiger Schwierigkeitsgrad
Das System soll in sinnvollen, kleinen Schritten eine Anpassung des Schwierigkeitsgrades durchführen.
- Sofortige Rückmeldung mit Kommentaren
Nach Bearbeitung einer Aufgabe und Übersendung des Lösungsvorschlages soll eine umgehende, didaktisch sinnvolle Rückmeldung erfolgen. Hierbei ist, soweit möglich, mindestens ein richtiger Lösungsweg aufzuzeigen. Die Bewertung soll darauf aufbauend nachvollziehbar sein.
- Integration von Lerngruppen
Die Möglichkeit zur Bildung und Unterstützung von Lerngruppen soll durch

4 Anforderungen an ein Kommunikations- und Tutoring System für Lerngruppen

das System ermöglicht werden. Hierbei soll insbesondere auch die Möglichkeit gegeben werden, sowohl die Gruppenleistung wie auch die individuelle Leistung zu bewerten und als Feedback an die Lerner zurück zu geben.

4.9.2 Organisatorische Anforderungen

Neben den Verbesserungen innerhalb des Beurteilungssystems sollen noch weitere kritische Punkte verbessert werden:

- **Gebrauchstauglichkeit**
Das System soll in bestehenden Infrastrukturen nutzbar sein. Dies bedeutet, dass ausschließlich zukunftsorientierte Standardtechnologien verwendet werden sollen.
- **Curriculare Übertragbarkeit**
Das System soll einerseits innerhalb einer Universität von unterschiedlichen Fachbereichen einsetzbar sein als auch andererseits an andere Universitäten übertragbar sein.
- **Begleitende Evaluation**
Das System soll während der Veranstaltung den Dozierenden eine regelmäßige Rückmeldung über den Lernfortschritt der Studierenden geben können.
- **Nachhaltigkeit**
Das System soll durch seine Architektur so flexibel sein, dass es ohne größeren Aufwand möglich ist, einen längerfristigen Betrieb sicherzustellen.
- **Integration in das bestehende Curriculum**
Das System soll möglichst leicht in bestehende Curricula integriert werden können. Dies bedeutet, dass das System keine grundlegenden Veränderungen in der Konzeption der existierenden Veranstaltung erforderlich macht, sondern zumindest als begleitendes Instrument jederzeit einsetzbar ist.
- **Positive Veränderung in der Organisationsentwicklung**
Das System soll bestehende organisatorische Strukturen nicht behindern, sondern, falls möglich, unterstützen.
- **Orts- und zeitunabhängiges Training**
Das System soll jederzeit und möglichst überall einsetzbar sein. Hierbei ist von

4.9 Allgemeine Verbesserungsmöglichkeiten

einer durchschnittlichen EDV-Ausstattung und einer durchschnittlichen Internetverbindung auszugehen.

4 Anforderungen an ein Kommunikations- und Tutoring System für Lerngruppen

5 Architektur des Communication and Tutoring System (CATS)

5.1 Grundkonzeption

Die Grundidee des im Rahmen dieses Promotionsvorhabens entwickelten Communication and Tutoring System (CATS) besteht in der Bildung von virtuellen Lernteams unter der Berücksichtigung unterschiedlicher Fachkenntnisse der Studierenden sowie in der Unterstützung der Kommunikation zwischen den Studierenden innerhalb der Lernteams.

Die Bildung der Lerngruppen geschieht in mehreren Phasen: In der ersten Phase (Szenario 1) erhalten die Studierenden die Möglichkeit, mit adaptiven Übungsaufgaben beliebig oft zu trainieren. In dieser Phase ist eine Internet-Verbindung nicht erforderlich. Die Aufgaben können auch über ein Offline-Medium (z. B. DVD) angeboten werden.

In einer zweiten Phase (Szenario 2), die eine Internetverbindung voraussetzt, können die Studierenden ihre Ergebnisse dem CATS-Server übermitteln. Dieser berechnet daraufhin in Echtzeit pro Aufgabe jeweils eine Rangliste und ermöglicht den Studierenden zudem eine Instant-Kommunikation zu anderen Studierenden, die Erfahrungen in der Bearbeitung der jeweiligen Aufgaben haben und im Moment im Internet erreichbar sind.

In der dritten Phase (Szenario 3) werden schließlich Lerngruppen gebildet, und diese erhalten jeweils wiederum in Abhängigkeit vom Kenntnisstand aller ihrer Mitglieder eine Gruppenaufgabe gestellt. Während der Bearbeitung der gemeinsamen Aufgabe wird durch CATS ebenfalls eine Gruppenkommunikationsunterstützung für die Studierenden angeboten.

5.2 Individuelle, adaptive Aufgaben: Szenario 1

Wie in Abschnitt 4.2 dargestellt, soll sich das System adaptiv an den Leistungsstand der Studierenden anpassen. Hierzu ist zunächst festzulegen, welche Kennzahlen sich aus der Bearbeitung einer Aufgabenstellung entnehmen lassen, die es ermöglichen, Rückschlüsse auf die Leistungsfähigkeit der Studierenden zu ziehen.

5.2.1 Kennzahlensystem zur Wissensbewertung

In CATS wird, wie in [LE03a] dargestellt, ein hierarchisches Kennzahlensystem verwendet, welches in aggregierter Form die Vergleichbarkeit der Studierenden sicherstellen soll. Die Kennzahlen sollen jedoch nicht nur im zeitlichen Ablauf erfasst und aufgezeichnet werden, sondern sie sollen auch eine Vergleichbarkeit mit Lernzielvorgaben ermöglichen. Solche Lernzielvorgaben können jeder Übungsaufgabe hinzugefügt werden. Die Lernziele selbst werden durch drei generische Grundparameter bestimmt, die sich aus der Item-Response-Theorie von Wainer [Wai90] ergeben. In Abschnitt 3.2.2 wurde bereits eine kurze Einführung in diese Theorie gegeben. Da die Lösungen zu den Aufgaben, im Gegensatz zu Multiple-Choice-Fragen, nur sehr schwer zu erraten sein sollten (da z. B. keine festen Lösungsvorschläge vorgeben werden), ist der Ratefaktor mit 0 anzusetzen. Ebenso sollten die Aufgaben über eine sehr hohe Trennschärfe bezüglich des Schwierigkeitsgrades verfügen, so dass sich bei *idealen* CATS-Aufgaben ein Abgrenzungsfaktor zu 1 ergibt. Damit würde in einem solchen Fall ein 1PLM oder auch als Rasch bezeichnetes Modell entstehen. Zu diesem Modell siehe [MN99]. Da die Studierenden auf die Abschlussklausur vorbereitet werden sollen, sind zudem Schwierigkeitsgrad, die benötigte Zeit, wie auch die Zuverlässigkeit von Interesse.

Diese Parameter sind im Einzelnen:

Schwierigkeitsgrad (Proficiency) Dieser Parameter kennzeichnet den Schwierigkeitsgrad einer bestimmten Aufgabe. Innerhalb des Designs einer einzelnen Aufgabe ist auf eine möglichst hohe Dynamik zu achten. Mit jeder schwierigeren Bearbeitungsstufe soll der Wert sich signifikant erhöhen. Dieses Konzept, welches aus dem Bereich der Computerspiele als so genannte "Spiel-Levels" wohl bekannt ist, wirkt sich motivierend auf die Benutzer aus.

In die Berechnung dieses Messwertes gehen meist mehrere weitere Parameter ein. So kann z. B. in Mathematikaufgaben eine höhere Stellenanzahl von Aus-

gangswerten die Komplexität und damit die Schwierigkeit der Rechenaufgabe erhöhen, oder in einer Informatik-Übung werden die zu verwendenden Algorithmen komplizierter. Diesen Sachverhalt hat der Ersteller der Aufgabe zu berücksichtigen und die entsprechenden Schwierigkeitsgrade zuzuweisen.

Während der Erstellung einiger Beispielaufgaben zeigte sich häufig ein linearer, quadratischer oder exponentieller Zusammenhang zwischen den einzelnen Eingabeparametern und dem gesamten Schwierigkeitsgrad der Aufgabe. Dies ergibt sich unter Anderem durch die notwendigen Lösungsschritte und die damit jeweils verbundene Komplexität in der Bearbeitung. Ein Beispiel soll dies illustrieren: Bei Additionsaufgaben steigt der Schwierigkeitsgrad durch eine erhöhte Stellenanzahl linear an, da lediglich in Abhängigkeit der Stellenzahl jeweils häufiger der Übertrag gebildet werden muss. Während hingegen die Lösung von Gleichungssystemen, etwa wenn der Lerner das Gaußsche Eliminationsverfahren anwenden soll, der kubischen Komplexitätsklasse angehört. Dieser Zusammenhang muss durch den jeweiligen aufgabenspezifischen Wertebereiche abgebildet werden. Hierbei werden von dem System Ganzzahlwerte verwendet, die für den einfachsten Aufgabentyp den Wert 1 annehmen sollen. Dieser Startwert dient als Norm für alle weiteren Schwierigkeitsgrade. Sie sollen entsprechend abgeschätzt werden. Ist in der nächste Stufe die Arbeit ungefähr etwa 10 fach schwieriger zu bearbeiten, so wäre der Wert dann für den Schwierigkeitsgrad 10. Der Wertebereich wird aus diesem Grund von dem System nicht begrenzt. Dem Autor der Aufgabe ist insoweit ein weiter Spielraum eingeräumt.

Zuverlässigkeit (Reliability) Die Zuverlässigkeit gibt an, wie oft die Studierenden Aufgaben innerhalb einer Sitzung richtig gelöst haben. Dieser Parameter erhöht sich für jede richtig gelöste Aufgabe und erniedrigt sich für jede falsch gelöste Aufgabe.

Zeitbedarf (Time) Der Zeitbedarf berechnet sich in Millisekunden. Der Zähler wird für jede neu generierte Aufgabe ebenfalls neu gestartet.

Diese drei Werte werden sowohl als Sollvorgaben (etwa zur Erfüllung der Examensreife) von dem Dozenten bestimmt, als auch verpflichtend von jeder CATS-Aufgabe gemessen. Diese drei Werte sind so genannte Key-Performance-Indikatoren (KPI). Sie geben im Laufe einer begleitenden Evaluierung den Fortschritt bezüglich der Lernzielerfüllung an.

5 Architektur des Communication and Tutoring System (CATS)

Am Ende einer Sitzung sollen nun neben den Authentifizierungsdaten (Kennung und Passwort) und Aufgaben-ID die Messwerte der jeweiligen Sitzung übertragen werden. Aus diesen Zahlen bestimmt das System einen Vergleichswert, um jeweils für jede Übungsaufgabe eine Rangliste zu erstellen. Dieser als *Performance-Indicator* bezeichnete Wert kann grundsätzlich frei berechnet werden. Im Moment wird folgende Ausgangsgleichung verwendet:

$$\text{Performance} = \frac{\alpha \cdot \text{Proficiency}}{\beta \cdot \text{Time}} \cdot \gamma \cdot \text{Reliability} \quad (5.1)$$

Um die verschiedenen Ranglisten vergleichbar zu machen, können die Werte α , β , γ je nach Anwendungsfeld, festgelegt werden, je nachdem wie wichtig im Verhältnis zueinander Schwierigkeitsgrad, Zuverlässigkeit und Zeit sind. So ist z. B. bei zeit- und sicherheitskritischen Aufgaben (z. B. ein Trainingssystem für Fluglotsen) eine hohe Gewichtung der Zeit und der Zuverlässigkeit vorzunehmen, während der Schwierigkeitsgrad eher untergeordnet ist. Demgegenüber würden bei juristischen Aufgaben der Schwierigkeitsgrad und die Zuverlässigkeit eine höherer Bedeutung als die benötigte Zeit haben. Da es sich um eine multiplikative Verknüpfung der KPIs handelt, haben die Faktoren jedoch auf die Reihenfolge innerhalb einer Rangliste keinen Einfluss.¹

Um eine entsprechende Abwägung, die meist nach subjektiven Kriterien vorgenommen werden wird, bietet sich z. B. die aus der Systemtheorie stammende *Nutzwertmethode* nach Zangenmeister [Zan76] an. Hierbei würde ausgehend von allgemeinen Lernzielen die Wichtigkeit der einzelnen Parameter eingeschätzt werden. Tabelle 5.1 stellt das entsprechende Gewichtungsschema dar. Ausgehend von der maximal er-

Sehr wichtig	100 Punkte
Wichtig	75 Punkte
Neutral	50 Punkte
Weniger wichtig	25 Punkte
Unwichtig	1 Punkt

Tabelle 5.1: Bewertung der Wichtigkeit einzelner KPIs

reichbaren Gesamtpunktzahl, werden anschließend die prozentualen Gewichte berechnet. Nach einigen Probeläufen mit verschiedenen Lernertypen, muss im Rahmen einer *Sensitivitätsanalyse* sichergestellt werden, dass leichte Veränderung an den

¹Die Werte α , β , γ ergeben ausmultipliziert lediglich einen konstanten Faktor.

Messwerten keine signifikanten Auswirkungen haben.

Der Schwierigkeitsgrad und die Zuverlässigkeit sorgen für einen hohen Wert und ein erhöhter Zeitbedarf reduziert wiederum den Wert. Durch den Quotienten ergibt sich bezüglich der Dimension ein Leistungsindikator, wie er auch bei Prüfungen von Bedeutung ist.

Der Performance-Indikator wird zurzeit in CATS wie folgt berechnet:

$$\text{Performance} = \frac{\text{Proficiency}}{\text{Time}} \cdot \text{Reliability} \quad (5.2)$$

Hierbei wurden die Gewichte zu 1 gesetzt. Denn die Studierenden sollen auf ihre Abschlussklausur vorbereitet werden. Insoweit sind alle drei Parameter von gleicher Wichtigkeit. Je höher der Wert ist, desto besser ist die Position auf der Rangliste.

5.2.2 Adaptivität

Bisherige adaptive Aufgaben wurden oft auf der Grundlage von Multiple-Choice-Aufgaben erstellt.

Die Adaptivität der CATS-Aufgaben wird dadurch sichergestellt, dass beim Neustart einer Aufgabe eine Aufgabenstellung in der niedrigsten Schwierigkeitsstufe erfolgt. Sobald diese Aufgabe korrekt bearbeitet wurde, wird sowohl der Schwierigkeitsgrad der nächsten Aufgabe erhöht als auch der Wert der Zuverlässigkeit. Scheitert allerdings der Benutzer an einer gegebenen Aufgabe, so wird der Schwierigkeitsgrad beibehalten und die Zuverlässigkeit um eins erniedrigt. Die Studierenden erhalten jedoch eine korrekte Lösung angezeigt (meist in Form einer Animation), um das Verständnis bezüglich der Bearbeitung der Aufgabe zu erhöhen. Anschließend erhalten sie nun eine andere Aufgabenstellung des gleichen Schwierigkeitsgrades. Hierzu werden bestimmte Parameter (z. B. gegebene Ausgangswerte) durch einen Zufallsgenerator neu belegt.

Durch dieses Verfahren wird sichergestellt, dass die Studierenden einerseits relativ schnell an ihre Leistungsgrenze geführt werden, andererseits wird pro Sitzung der jeweils zu dem Zeitpunkt vorhandene Kenntnisstand ermittelt. Damit ist es auch möglich, zeitliche Schwankungen pro Lernziel zu analysieren.

5.2.3 Realisiertes Szenario 1: Adaptive, individuelle Übungsaufgaben

Sehr viele Studierende verfügen heute immer noch nicht über einen breitbandigen Internetanschluss. Dies hängt zum einen mit der fehlenden technischen Infrastruktur bzw. mit den Kosten, die mit solch einem Anschluss verbunden sind, zusammen. Aus diesem Grunde enthält das CATS-Konzept ein Szenario, bei dem die adaptiven Übungsaufgaben auch im "Offline-"Modus von den Studierenden benutzt werden können. Die Distribution der Aufgaben kann dann durch eine CD oder DVD erfolgen.

Das Szenario wird in Abbildung 5.1

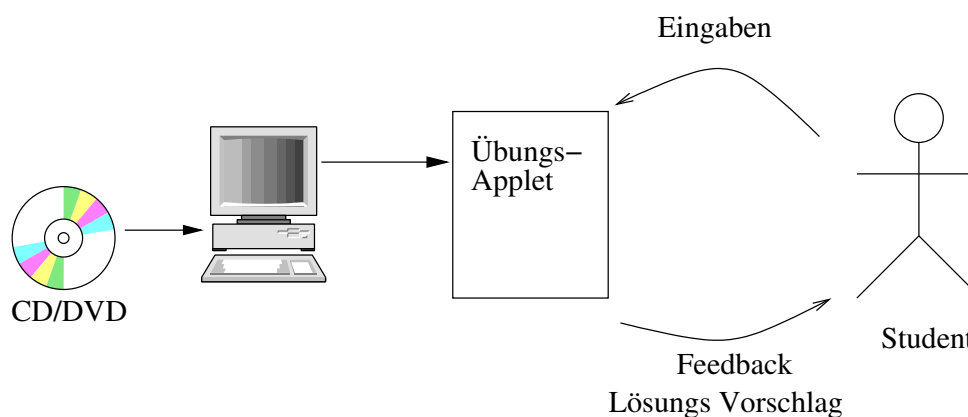


Abbildung 5.1: Offline-Szenario von CATS

schematisch dargestellt.

Hierbei erhalten die Studierenden nach Eingabe ihrer Lösung ein sofortiges Feedback, ob ihre Lösung richtig ist. Im Fehlerfall erfolgt die Demonstration des Lösungsweges mit entsprechenden Hinweisen.

5.3 Die Peer-to-Peer Kommunikationsanbindung in CATS: Szenario 2

Die direkte Kommunikation zwischen den einzelnen Studierenden ist, wie bereits in Abschnitt 2.5.3 dargestellt, bei Fernstudierenden nur sehr eingeschränkt möglich. Aus diesem Grund unterstützt CATS die direkte, elektronisch vermittelte Kommunikation zwischen den Studierenden (siehe auch [LE03b]). Hierbei wurde das Internet

Abbildung 5.2 stellt die Protokollbündel dar. Die Norm Q.931 regelt den Verbindungsaufbau; um die Benutzung von virtuellen Verbindungen zu ermöglichen, sorgt die Norm H.225 für die Verhandlung und Kontrolle unter den Teilnehmern. Die Organisation der Registrations- und Statusinformationen (Registration Admission Status (RAS)) obliegt dem Protokoll H.225. Dieses gewinnt insbesondere bei der Verwendung von Infrastrukturen (z. B. Gatekeeper) an Bedeutung. Der Medientransport erfolgt für Audio über die G.7xx-Familie (z. B. G.700, G.728), für Video über die H.26x-Familie (H.261, H.263, H.264). Die Datenströme sind ausgehend von komprimierten Bildern schichtenweise aufgebaut. Als Transportprotokoll wird das Realtime Protocol (RTP)[SCFJ96a] verwendet. Es kapselt die Datenpakete ein und versieht sie mit zusätzlichen Informationen über den verwendeten Medienstrom sowie die verwendete Zeitbasis (timestamping). Die Kontrolle erfolgt über das Real-Time Protocol Control Protocol (RTCP)[SCFJ96a].

Für die Steuerung von weiteren Applikationen (z. B. Shared Whiteboard) wird das Protokoll T.120 verwendet.

Verbindungsaufbau über H.323

Die Kommunikation über H.323 erfolgt in folgenden Schritten:

1. Registrierung der Teilnehmer
Über das Protokoll H.225 (RAS) werden Registrierungsdaten zum Server übertragen.
2. Initiierung des Kontaktes
Hierbei wirken die Protokolle H.225 und Q.931 zusammen.
3. Parameterverhandlung
Entsprechend den einzelnen Fähigkeiten der Endeinrichtungen findet nun eine Verhandlung über die beste Übertragungsmöglichkeit statt. Dies erfolgt entsprechend dem Protokoll H.245.
4. Initiierung der Kanäle zur Medienübertragung
Gemäß dem Protokoll H.245 werden nun die einzelnen Kanäle ausgehandelt und initiiert.
5. Medientransport
Die Mediendaten werden gemäß den Protokollen für Audio (G.7xx) und/oder

5.3 Die Peer-to-Peer Kommunikationsanbindung in CATS: Szenario 2

Video (H.26x) nach den Echtzeittransportprotokollen RTP/RTCP kodiert und übertragen.

6. Beendigung des Gespräches

Die Signalisierung der Beendigung des Gespräches erfolgt nach den Protokollen H.245, H.225 und RAS.

Realisierung in CATS

Die H.323-Anbindung wurde dadurch erreicht, dass sich die Studierenden, falls sie für andere Studierende erreichbar sein wollen, über eine Eingabemaske im CATS-System zunächst authentifizieren müssen. Bei diesem Vorgang wird ihre aktuell gültige IP-Adresse dem CATS-Server mitgeteilt. Des Weiteren haben sie eine H.323-konforme Applikation im Hintergrund zu aktivieren (z. B. Netmeeting), so dass sie, falls ein Anruf getätigt wird, auch erreichbar sind. Der CATS-Server zeigt nun pro angewählte Aufgabe alle verfügbaren Studierenden und deren Kenntnisstand an. Hierbei wird die proprietäre Universal Resource Identifier (URI) "callto" verwendet.

Wenn die Studierenden sich mit ihren Kommilitonen in Verbindung setzen möchten, klicken sie auf den entsprechenden Link, der Browser aktiviert die H.323-Applikation und übergibt die vom CATS-Server übermittelte IP-Adresse. Die Verbindung wird, wie oben dargestellt, aufgebaut, und die Studierenden können dann, per Audio-/Videokonferenz, Shared Whiteboard und Application Sharing untereinander kommunizieren und sich über die jeweilige Aufgabe austauschen.

5.3.2 Das Session Initiation Protocol SIP

In CATS wird außerdem auch die Kommunikation zwischen den Teilnehmern über das Session Initiation Protocol (SIP) realisiert.

Die SIP-Protokoll-Grundlagen

Das SIP-Protokoll wurde im RFC 2543[HSS⁺01] von der Internet Engineering Task Force (IETF) spezifiziert. Der Schwerpunkt dieses Protokolls liegt nicht wie bei H.323 in der kompletten Abwicklung der Kommunikationsanbindung, sondern es dient ausschließlich der Signalisierung, also die Lokalisierung und die Weiterleitung der

Verbindungsanforderung an den gesuchten Kommunikationspartner. Die Anwendung entscheidet darüber, ob dabei das Transmission Control Protocol (TCP) oder das User Datagram Protocol (UDP) verwendet wird. Allerdings müssen alle Infrastrukturelemente zumindest UDP unterstützen. Die Nutzdaten, in Form von Medienströmen, werden dann über das Real-Time-Transport Protocol (RTP) versendet [SCF]96b].

Nachrichten-Definition

Das SIP-Protokoll ist nachrichtenbasiert. Es werden sechs Nachrichten unterschieden:

- INVITE
Durch diese Nachricht wird einem anderen Benutzer mitgeteilt, dass ein Sitzungswunsch besteht.
- ACK
Diese Nachricht dient der Bestätigung des Verbindungsaufbaus.
- BYE
Der Gegenseite wird durch diese Nachricht der Wunsch nach Beendigung der Sitzung mitgeteilt.
- CANCEL
Diese Nachricht dient dem Abbruch eines bestehenden Verbindungsaufbauwunsches.
- OPTIONS
Diese Nachricht fordert die Server oder andere User Agents auf, ihre Fähigkeiten mitzuteilen.
- REGISTER
Durch diese Nachricht wird der Registrationsprozess beim Server vorgenommen. Mithilfe der Call Processing Language (CPL) kann zudem die Weiterleitungspolitik des Benutzers übermittelt werden.

Die Response-Codes werden durch eine dreistellige Nummer kodiert:

5.3 Die Peer-to-Peer Kommunikationsanbindung in CATS: Szenario 2

- 1.- Informational: Statusinformationen (z. B. Bearbeitung eines Verbindungswunsches).
- 2.- Success: Die Anfrage war erfolgreich.
- 3.- Redirection: Die Anfrage wird an eine andere Adresse weiter geleitet.
- 4.- Client Error: Die Anfrage war fehlerhaft (z. B. müssen die Authentifizierungsdaten erneut übertragen werden).
- 5.- Server Failure: Der Server ist derzeit in einem fehlerhaften Zustand.
- 6.- Global Failure: Kein Server ist in der Lage, die Anfrage zu bearbeiten.

SIP-Infrastruktur-Elemente

Die SIP-Infrastruktur besteht aus folgenden Elementen[HSS⁺01], [JLSS01]:

SIP User Agent Als User Agents werden alle die Endgeräte bezeichnet, die das SIP-Protokoll unterstützen.

SIP Registrar Server Die Registrar Server dienen dazu, dass sich die User Agents mit ihrer SIP-Adresse und ihrer IP/Port-Nummer dort registrieren. Durch diesen Server werden die SIP-Adressen den jeweils gerade für den Benutzer gültigen IP/Port-Adressen zugewiesen.

SIP Proxy Server Die SIP Proxy Server sind vergleichbar mit klassischen Telefonanlagen, die eine komplexere Telefon-Weiterleitungspolitik beherrschen. Sie nutzen den Registrar Server zwar zur Lokalisierung der Benutzer, ermöglichen aber zudem die Gruppen-Signalisierung an mehrere Teilnehmer und die finale Gesprächszuordnung.

SIP Redirect Server Dieser Server erlaubt die Weiterleitung eingehender Signalisierungen an andere Teilnehmer.

Realisierung der SIP-Anbindung in CATS

In CATS wurde eine SIP-Kommunikationsunterstützung realisiert. Im Rahmen einer Studienarbeit [Bei03] wurde die bestehende Infrastruktur erweitert. Zunächst wurde in der Datenbank des CATS-Servers die Tabelle der Nutzer um ein Feld, der SIP-Adresse, erweitert. Abbildung 5.3 stellt die Anbindung schematisch dar: Die Kom-

5 Architektur des Communication and Tutoring System (CATS)

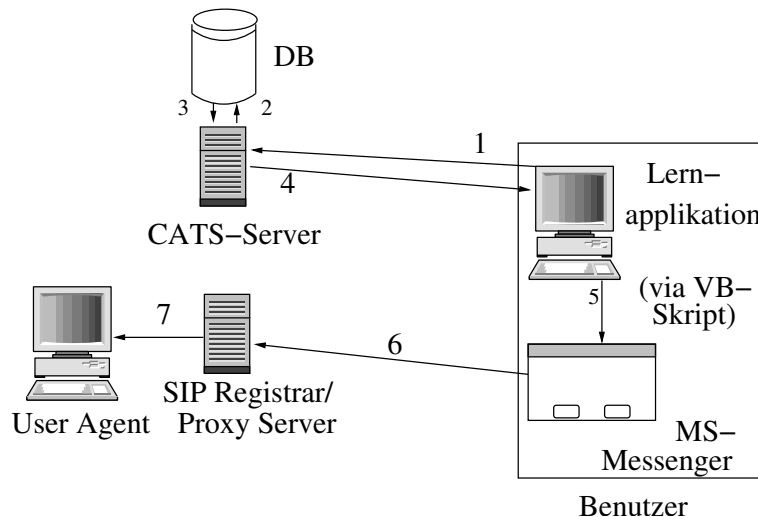


Abbildung 5.3: Die SIP-/CATS-Integration nach [Bei03][S.29ff.]

munikationsanfrage kann entweder den oder die zurzeit Beste/Besten bezüglich einer bestimmten Aufgabe, die im Moment erreichbar sind, betreffen, oder einen bestimmten Studierenden.

Die Suche beginnt ausgehend von der Lernapplikation (Schritt 1), die Anfrage wird an den CATS-Server übermittelt. Dieser wertet die Datenbank aus (Schritt 2), die SIP-Adresse des verfügbaren Adressaten wird an den CATS-Server übermittelt (Schritt 3), diese Adresse wird der Lernapplikation mitgeteilt (Schritt 4). Die Lernapplikation wiederum übermittelt die SIP-Adresse nun durch ein Visual-Basic-Skript dem MS Messenger, welcher wie bereits in Abschnitt 3.9.2 erwähnt mittlerweile bei allen neu ausgelieferten MS-Windows-Systeme verfügbar ist. Der MS-Messenger startet nun die Verbindungsanfrage an den SIP Registrar/Proxy Server (Schritt 6), dieser stellt dann die Verbindung zu dem User Agent des gesuchten Teilnehmers gemäß dem SIP-Protokoll her (Schritt 7). Im Rahmen der Studienarbeit wurde auch ein SIP-Registrar-Server eingerichtet.

Die Studierenden können sich nun über ihre SIP-Kommunikationseinrichtungen unterhalten und Informationen über die jeweiligen Aufgaben austauschen.

5.3.3 Bewertung und Testat der individuellen Leistungsbeiträge

Da eines der Ziele von CATS die erfolgreiche Vorbereitung auf eine Abschlussprüfung ist, sollen die Studierenden jederzeit Informationen über ihren individuellen

Leistungsstand haben. Das System stellt hierbei ein kontinuierliches Self-Monitoring zur Verfügung. Hierbei ermöglicht es CATS, durch geeignete Vorwahl der "Soll-Parameter" für die einzelnen CATS-Übungsaufgaben die Studierenden jederzeit über ihre individuelle Prüfungsreife zu informieren.

Andererseits kann es in den ersten Lernphase, in der das Wissen der Studierenden im Verhältnis zum Prüfungsziel in aller Regel noch sehr gering ist, sinnvoll sein, die Studierenden über den individuellen *Lernfortschritt* zu motivieren. Gerade in dieser ersten Phase lassen sich schnell große Lernfortschritte erzielen. Dadurch ist eine positive Motivation möglich. Aus diesem Grund erhalten auch die Studierenden eine Rückmeldung, die ausgehend von ihren bisherigen Ergebnissen die besten Ergebnisse berechnet und sie mit denen der anderen Studierenden vergleicht. Es erfolgt als Rückmeldung die aktuelle Rangposition innerhalb aller Teilnehmer, die diese Aufgabe bearbeitet haben.

5.3.4 Realisiertes Szenario 2: Unterstützung der Kommunikation zwischen Lernern

Wenn die Studierenden nur über eine schmalbandige Internetverbindung verfügen (z. B. Analog-Modem), können sie zumindest ihre Leistungsdaten zu dem zentralen CATS-Server übertragen. Abbildung 5.4 stellt die beiden Rückkopplungsschleifen dar. Einerseits die aus dem 1. Szenario bekannte Rückkopplungsschleife mit Musterlösung, andererseits nun die Rückkopplung durch das CATS-System. Hierzu werden zunächst die Daten abgespeichert. Anschließend erfolgt eine Korrelation mit weiteren, ebenfalls in der Datenbank vorhandenen Daten.

Sie erhalten nun nicht mehr nur ein einfaches Feedback über ihre Lösung, sondern auch eine Rückmeldung, die frei gestaltet werden kann. Im Probetrieb war dies die eigene, aktuelle Rangposition. Die Rückmeldung kann aber auch, falls die Studierenden in Gruppen zusammengefasst wurden, über eine etwaige Gruppenrangposition informieren wie auch z. B. den Erfüllungsgrad bestimmter Lernziele.

Insoweit die Studierenden über einen breitbandigen Internetzugang verfügen, können sie die Audio-/Video-Kommunikationsunterstützung verwenden. Die Studierenden haben nun die Möglichkeit, ihre Bereitschaft, für Auskünfte zur Verfügung zu stehen, an den CATS-Server zu übermitteln. Dieser wertet die Daten aller Studierenden aus und visualisiert in entsprechenden Übungsumgebungen, die gerade im System vorhandenen Studierenden. Diese werden in der Reihenfolge ihrer indi-

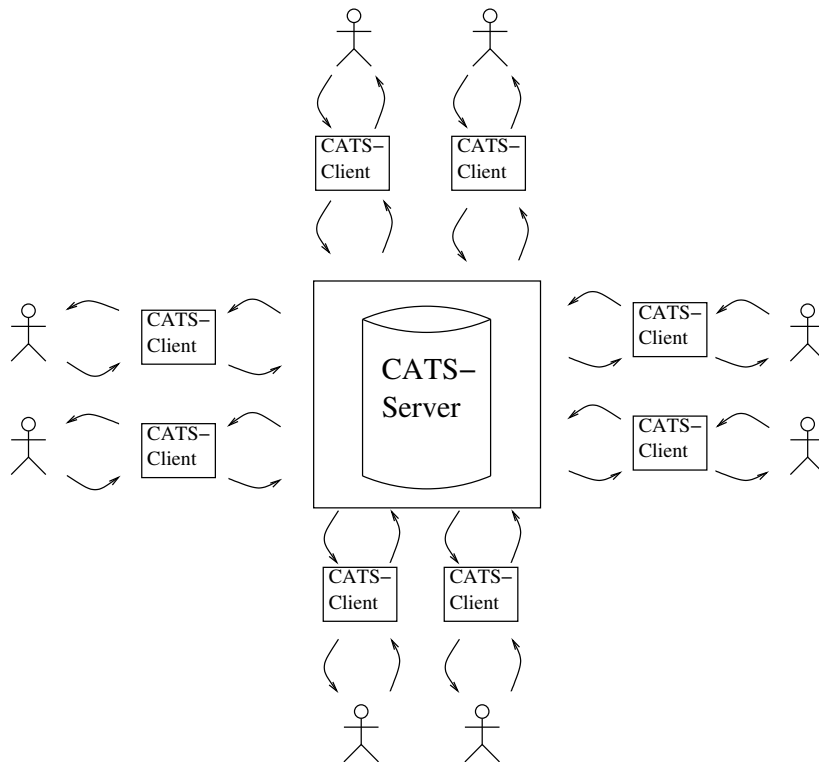


Abbildung 5.4: Online-Übung mit zweifacher Rückkopplung

viduellen Kompetenz aufgelistet. Ein Nebeneffekt hierbei ist, dass eine Awareness zwischen den Studierenden hergestellt wird, der insbesondere in der Fernlehre in dieser Form so bisher nicht gegeben war.

5.4 Unterstützung von Lernergruppen durch CATS: Szenario 3

Ein Schwerpunkt des CATS-Systems bildet die in [LE04a] vorgestellte Gruppenarbeitskomponente. Hierbei sollen insbesondere rollenorientierte Gruppenarbeiten ermöglicht werden. Hierfür ist es zu Lösung einer konkreten Aufgabenstellung erforderlich, bestimmte Rollen zu erfüllen. Erst wenn alle Funktionen, die mit einer Rolle verbunden sind, korrekt ausgeführt werden, kann die Aufgabe gelöst werden. Die jeweiligen Rollen werden jeweils den Studierenden aufgrund ihrer Kenntnisse zugewiesen.

5.4.1 Das CATS-Auswahlverfahren zur Gruppenbildung

In Abschnitt 2.5.1 wurde u. a. dargelegt, dass eine hohe Ressourceninterdependenz zwischen den einzelnen Teilnehmern eine wichtige Voraussetzung für erfolgreiche Gruppenarbeit ist. Dies bedeutet, dass Wissen, welches ein Studierender hat, von einem anderen zur Erfüllung der Aufgabe unbedingt notwendig ist. Dadurch kommt es dann, wie in Abschnitt 2.4 dargestellt, zur Wissensvermittlung. Je stärker diese Interdependenz ist, desto stärker werden Kommunikationsprozesse initiiert. Aus diesem Grunde erfolgt die Gruppenzusammenstellung in CATS nach der Maßgabe einer möglichst hohen *Wissensdifferenz* zwischen den einzelnen Gruppenmitgliedern. Sobald die Studierenden alle Aufgaben einer Veranstaltung bearbeitet haben, liegen dem System die vollständigen Informationen über die Lernleistung eines jeden Studierenden über alle Lernziele vor. Ebenso ist im System festgehalten, wie gut ein Ziel erfüllt sein muss, um als ausreichend zu gelten.

Nun werden die Gruppen vom CATS-System automatisch nach zwei Regeln gebildet:

1. Jede Gruppe muss über das gesamte Wissen verfügen, welches zur Erfüllung der Aufgabe erforderlich ist.
2. Die Wissensdifferenz zwischen den Studierenden innerhalb der Gruppe soll möglichst hoch sein.

Die Abbildungen 5.5 und 5.6 stellen die Teambildung schematisch dar. Durch die
















Lernziele	A	B	C	D	E
Student 1					
Student 2					
⋮					
Student n					

Abbildung 5.5: Informationsübersicht Wissen pro Lernziel pro Studierenden

5 Architektur des Communication and Tutoring System (CATS)

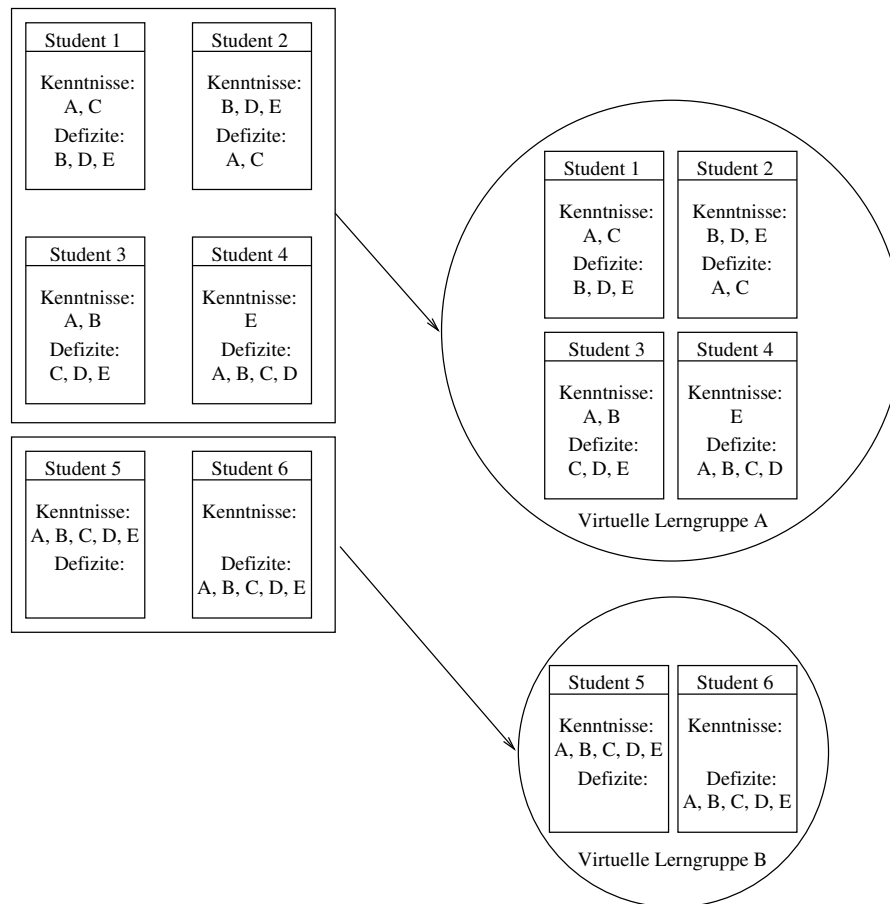


Abbildung 5.6: Bildung von Teams

Verfahren, die in Abschnitt 5.2.1 ermittelt und im Szenario 2 (s. Abschnitt 5.3) an das CATS-System übermittelt wurden, verfügt dieses nun über die Information darüber, wie stark die einzelnen Studierenden die Lernziele einer Veranstaltung erfüllen. Dies wird in Abbildung 5.5 schematisch dargestellt. Die Lernziele wurden in der Abbildung von A bis E durchnummeriert. Diese Information wird nun zur Gruppenzusammenstellung genutzt. In Abbildung 5.6 ist auf der linken Seite die Gesamtheit aller Studierenden eines Kurses und ihre Kenntnisse pro Lernziel dargestellt. Aus dieser Gesamtheit werden nun in diesem Beispiel die zwei Lerngruppen gebildet, die jeweils die höchsten Wissensdifferenzen haben; aber dennoch verfügt jede Gruppe über das Wissen, welches zur Erfüllung der Aufgabe notwendig ist.

Wenn das Verfahren auf die gesamte Lerngemeinschaft angewandt worden ist, ergibt sich nun eine Struktur, wie sie in Abbildung 5.7 dargestellt ist. Jede Arbeitsgruppe erhält eine individuelle, an die Leistungsfähigkeit ihrer Mitglieder angepas-

Kurs/Klasse

	1	2	3	...	n
Arbeitsgruppe 1					
Rolle/Untergruppe 1					
Rolle/Untergruppe 2					
⋮					
Rolle/Untergruppe n					

Abbildung 5.7: Aufteilung der Lerngemeinschaft in Arbeitsgruppen und Rollen bzw. Untergruppen

te Aufgabe. Innerhalb dieser Arbeitsgruppe besteht nun die Möglichkeit, entweder den einzelnen Gruppenmitgliedern direkt spezifische Rollen zuzuordnen oder selbst noch einmal Unterarbeitsgruppen pro Rolle zu definieren.

5.4.2 Die Erstellung einer adaptiven Gruppenaufgabe

Nach der Bildung der Lerngruppen erfolgt nun die Generierung einer adaptiven Gruppenaufgabe. Hierzu werden die Maxima der Studierenden über alle Lernziele innerhalb der Gruppe betrachtet und daraus wird ein Gruppen-Schwierigkeitsgrad berechnet. Dieser Vorgang ist in Abbildung 5.8 dargestellt. In dieser Annahme seien

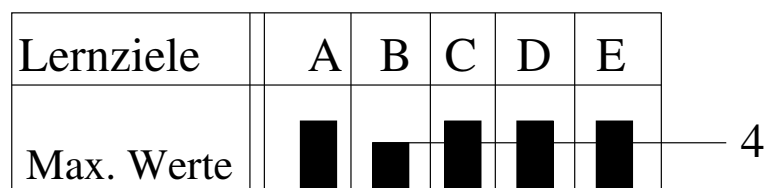


Abbildung 5.8: Berechnung des Gruppenschwierigkeitsgrades

die erreichten maximalen Werte pro Lernziel entsprechend des Proficiency-Grades von 5 für die Lernziele A, C, D, E erreicht worden und für das Lernziel B der Wert 4.

Damit kann der Gruppe eine Aufgabe bis zu dem Schwierigkeitsgrad 4 gestellt werden, denn es befindet sich mindestens ein Studierender in der Gruppe, der diesen Schwierigkeitsgrad bewältigen kann.

Dieser Wert wird dann im Rahmen des Initiierungsprozesses an das Gruppenobjekt übergeben. Es wird daraufhin eine konkrete, entsprechend parametrisierte Aufgabenstellung generiert. Ein Beispiel aus dem Bereich der Rechtswissenschaft soll diese Aufgabengenerierung illustrieren: Die Schwierigkeitsstufe 1 entspricht laienhaften Rechtskenntnissen, die Stufe 2 entspricht Jura-Kenntnissen, die nach dem Grundstudium zu erwarten wären, die Stufe 3 werden nach dem Hauptstudium erwartet, die Stufe 4 entspricht dem Schwierigkeitsgrad des 1. Staatsexamens, die Stufe 5 entspricht dann dem Schwierigkeitsgrad des 2. Staatsexamens.

Aus einem Pool von möglichen Grundfällen werden nun entsprechende Varianten generiert. Hierbei könnte in der 4. Schwierigkeitsstufe z. B. im Strafrecht jedes Merkmal der Strafbarkeit (siehe hierzu Abschnitt 7.3) bis zur schwierigsten Stufe geprüft werden. Prozessuale Fragestellungen sind jedoch, im Gegensatz zur Schwierigkeitsstufe 5, und damit dem 2. Staatsexamen, nur von einfacher Gestalt.

Im nächsten Schritt werden dann den Studierenden die jeweiligen Rollen zugewiesen. In diesem Beispiel einer simulierten Gerichtsverhandlung wären dies die Rollen Staatsanwalt, Verteidiger und Richter.

Hierbei wird vom CATS-System ein generisches Framework vorgegeben. Dies bedeutet, dass von dem Gruppenarbeitsobjekt nur die Schnittstellen eingehalten werden müssen und die konkrete Ausgestaltung der Gruppenarbeit von dem CATS-System nicht beeinflusst wird. So wäre es denkbar, dass innerhalb des Gruppenobjektes auch Themenbereiche aus anderen Fächern realisiert werden. So könnten z. B. auch so genannte Participatory Simulations, etwa in Form einer Börsensimulation, als inneres Gruppenobjekt programmiert werden. Der Proficiency-Level würde hier der Komplexität der Börse entsprechen, einfache Güter, mehrere Güter mit einer gegenseitigen Substitutionsmöglichkeit oder gar die Integration von derivativen Finanzinstrumenten.

5.4.3 Die Gruppenarbeitsarchitektur

Abbildung 5.9 stellt das Gruppenarbeitsobjekt schematisch dar. Das Objekt verfügt über zwei Elemente, einem äußeren Rahmen, der für alle CATS-Gruppenobjekte gleich ist und insbesondere dem CATS-Server zur Verwaltung dient, und einem in-

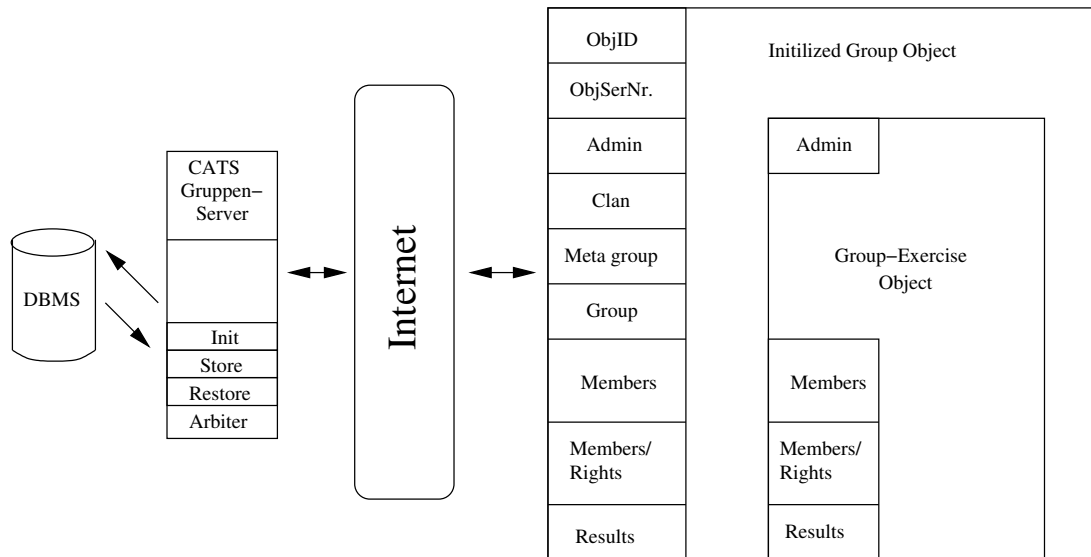


Abbildung 5.9: Das CATS-Gruppenarbeitsobjekt

neren Übungsobjekt. Nur dieses Übungsobjekt muss auf die jeweilige Aufgabenstellung angepasst werden. Hierbei sind die vorgenannten Kriterien bezüglich des Schwierigkeitsgrades zu beachten.

Die generische Verwaltung erfolgt dann durch ein Rollenkonzept, wobei jede Rollen durch ein Bündel von Rechten ausgedrückt wird. Dies bedeutet, dass eine Rolle auf bestimmte Objekte zugreifen und gegebenenfalls beeinflussen kann.

Am Beispiel eines simulierten Moot-Courts soll dieses Konzept näher erläutert werden. Abbildung 5.10 stellt diesen Sachverhalt noch zusätzlich schematisch dar. Das Gruppenübungsobjekt besteht in diesem Fall aus vier Hauptobjekten: Der Staatsanwaltschaft, der Verteidigung, dem Gericht und dem Reviewer (als neutraler Beobachter). Die Objekte der Staatsanwaltschaft, Verteidigung und Gericht besitzen wiederum jeweils vier Unterobjekte, wie sie auch der juristischen Arbeitsweise entsprechen: Einem Brainstormingobjekt, einem Gliederungsobjekt, einem Gutachtenobjekt sowie einem Schriftsatzobjekt. Bei dem Review-Objekt, welches der Beurteilung des Prozesses durch einen Experten dient, sind neben den individuellen Beurteilungen der einzelnen Beteiligten, auch eine Gesamtbewertung der Verhandlung vorgesehen. Ein Lerner, der in einer virtuellen Gerichtsverhandlung die Rolle eines Staatsanwaltes einnimmt, kann zwar auf alle Objekte der Staatsanwaltschaft vollständig zugreifen (sowohl lesend, wie auch schreibend), auf die Objekte etwa der Verteidigung erfolgt nur ein eingeschränkter Zugriff. In diesem Beispiel könnte die Staatsanwalt-

5 Architektur des Communication and Tutoring System (CATS)

Sachverhalt: Öffentlich

Staatsanwaltschaft			Verteidigung		
Rechte	Elemente		Rechte	Elemente	
L,S	Brainstorming	vertraulich	L,S	Brainstorming	vertraulich
L,S	Gliederung		L,S	Gliederung	
L,S	Gutachten		L,S	Gutachten	
L,S	Schriftsätze	öffentlich	L,S	Schriftsätze	öffentlich

Gericht				
Rechte	Elemente		Rechte	Reviewer
L,S	Brainstorming	vertraulich	L,S	Beurteilung S
L,S	Gliederung		L,S	Beurteilung V
L,S	Gutachten		L,S	Beurteilung G
L,S	Schriftsätze	öffentlich	L,S	Allgemeine Beurteilung

Abbildung 5.10: Die Organisation der Gruppenrechte am Beispiel des Moot-Courts

schaft etwa nur auf das Objekt "Schriftsätze" lesend zugreifen. Die Objekte Brainstorming, Gliederung und Gutachten hingegen bleiben den Mitgliedern der Rolle "Staatsanwalt" vollständig verborgen.

Schließlich existiert noch die Möglichkeit, die Ergebnisse der Aufgabe, wie bereits bei den individuellen Übungsaufgaben, an den CATS-Server zu übermitteln.

5.4.4 Die Gruppenkommunikation

Für die Audio-/ Videokommunikation mit mehr als zwei Teilnehmern stehen mehrere Möglichkeiten offen: Zum einen könnte man IP-Multicast benutzen, eine Gruppenadresse verwenden und mithilfe der verschiedenen Mbone-Tools eine Audio-Video-Konferenz initiieren, zum anderen kann man im Rahmen der H.323-Technik eine Multipoint Control Unit (MCU) verwenden. SIP-basierte Verfahren waren ausgeschlossen, da zu dem Zeitpunkt der Realisierung der Kommunikationsinfrastruktur dieser Arbeit weder Produkte für Videokonferenzen per SIP, noch Konferenzsysteme mit mehr als zwei Teilnehmern zu Verfügung standen. Dies hat sich in

der Zwischenzeit geändert, fast alle Videokonferenzsysteme, die neu auf den Markt kommen, unterstützen nun auch SIP. Außerdem gibt es mehrere Produkte bzw. Projekte die die Integration von bestehenden VoIP-Systemen und zum Teil auch Videokonferenzsysteme (z. B. Cisco IOS Software Release 12.2(2)XB [CS05]) zum Ziel haben. Da Multicast-IP selbst nicht von vielen Internet Service Providern (ISP) unterstützt wird und daher eine entsprechende Anbindung der Studierenden nicht durchgängig möglich wäre, wurde im Rahmen dieser Arbeit eine H.323 basierte MCU eingesetzt. Diese ermöglicht insbesondere durch ihre Strukturierung in Räume die Verwendung in einem komplexeren Kommunikationsschema, welches im Folgenden detailliert besprochen wird.

Die Integration einer Multipoint Control Unit

Um auch Gruppen von Studierenden eine elektronische Konferenzmöglichkeit zu geben, ist es notwendig, soweit man ein H.323-konformes Kommunikationsprotokoll verwendet, eine so genannte Multipoint Control Unit (MCU) zu verwenden. Eine MCU ermöglicht die Bündelung mehrerer Audio- und Videoströme und überträgt dann diesen gebündelten Medienstrom zu den einzelnen Teilnehmern.

Die MCU stellt den Teilnehmern beliebig viele Konferenzräume zur Verfügung. Um zu einem bestimmten Konferenzraum zu gelangen, ist es notwendig, diesen in der von E-Mail-Adressen her bekannten Notation zusätzlich zur MCU-IP anzugeben. Dieser Gruppenname ist ebenfalls im CATS-System hinterlegt, so dass nach entsprechender Authentifizierung die entsprechenden Links automatisiert generiert werden können. Es ist jedoch hierbei zu beachten, dass gerade feste Gruppen sich bei dieser Art der Konferenz zu einem festen Termin verabreden müssen. Dies schränkt zwar zunächst die Zeitunabhängigkeit in der Kommunikation. Andererseits soll es aber gerade mehreren Studierenden ermöglicht werden, sich auszutauschen. Eine Möglichkeit, diese Einschränkung zu reduzieren, besteht in der möglichen, mehrdimensionalen Rollen-Kommunikation.

Die mehrdimensionale Rollen/Raum-Kommunikation

Eine Besonderheit des CATS-System ist die mehrdimensionale Gruppenkommunikation, wie sie in Abbildung 5.11 dargestellt wird. Ein Lerner gehört mit seiner Rolle einer Arbeitsgruppe an. Diese besteht aus Mitgliedern mit unterschiedlichen Rollen. Die Gruppen-Kommunikation ist nicht nur organisatorisch für die jeweilige Arbeits-

5 Architektur des Communication and Tutoring System (CATS)

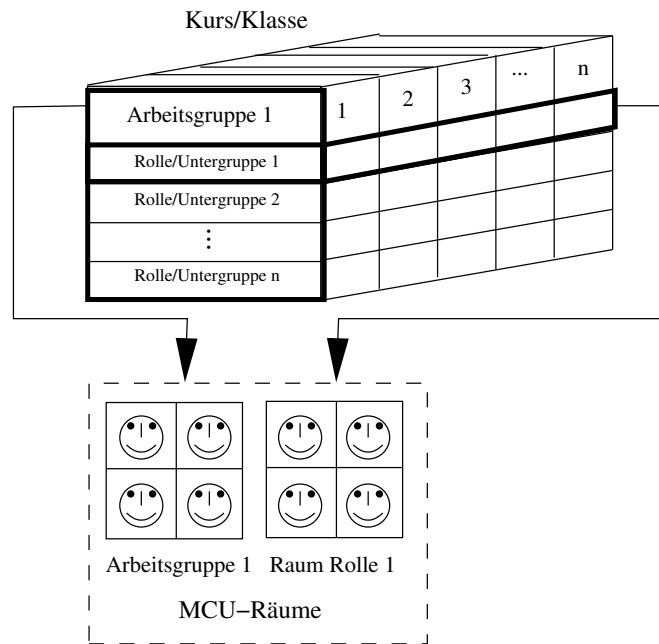


Abbildung 5.11: Die mehrdimensionale Gruppenkommunikation

gruppe, die eine konkrete Aufgabenstellung bearbeitet, möglich, sondern auch quer über alle Rollen hinweg. Dies entspricht auch dem in Abschnitt 2.4.2 vorgestellten Konzept der kollektiven Reflexion, basierend auf der Lerntheorie des sozialen Konstruktivismus (s. Abschnitt 2.3.3).

Es kann allerdings auch sinnvoll sein, einen Gruppendialog innerhalb einer "Rollen-Gemeinschaft" anzubieten. Ein Beispiel soll dies erläutern:

In einer simulierten Strafgerichtsverhandlung (so genannter Moot-Court) werden die drei wesentlichen Rollen Staatsanwalt, Verteidiger und Richter mit Studierenden besetzt. Im Laufe des simulierten Verfahrens kann es für die jeweiligen Rolleninhaber interessant sein, sich auch mit anderen Personen, die die gleiche Rolle innehaben, auszutauschen (z. B. alle Staatsanwälte tauschen Informationen untereinander aus). Dies entspricht auch insofern der Realität, dass in den meisten Fällen auch Organisationen (z. B. Fachverbände) existieren, die diese Kommunikationsprozesse unterstützen. Diese klären rollenspezifische Fragestellungen und ermöglichen eine entsprechende Weiterentwicklung.

5.4.5 Möglichkeiten der Gruppenbewertung

Neben der Individualbewertung soll CATS auch grundsätzlich eine Bewertung der Gruppenaufgabe ermöglichen. Hierbei gibt es ebenfalls mehrere Möglichkeiten der Auswertung. Eine Möglichkeit wäre, die Bewertung proportional zu der Summe aller absoluten Einzelbeiträge der Gruppenmitglieder vorzunehmen:

$$\frac{\sum_{k=1}^n \text{Endwert}_k}{n} \quad (5.3)$$

und andererseits eine relative Gruppenbewertung. Hierbei wird die Wissenssteigerung innerhalb der Gruppe bewertet:

$$\frac{\sum_{k=1}^n (\text{Endwert}_k - \text{Anfangswert}_k)}{n} \quad (5.4)$$

Wie bereits oben beschrieben, sollen die Gruppen nach Erkenntnissen der Pädagogischen Psychologie heterogen zusammengesetzt werden, mit dem Ziel einer möglichst intensiven Kommunikation über das Wissensgebiet. Aus diesem Grunde kann durch CATS der relative Wissenszuwachs bewertet werden, denn dies ist der Hauptmotivator und sollte auch entsprechend kommuniziert werden (z. B. durch entsprechenden Aushang). Durch die relative Gruppenbewertung lässt sich einerseits der Lernerfolg messen, andererseits lässt sich diese Bewertung auch zur Motivation z. B. im Vergleich zu anderen Gruppen (z. B. durch Gruppenranglisten) einsetzen. Hierdurch ist ein signifikanter didaktischer Mehrwert zu erwarten.

Eingang der Gruppenleistung in die Leistungsbeurteilung

Entsprechend den Einzelbeiträgen wird nun auch fortlaufend eine Bewertung vorgenommen. Da aus rechtlichen Gründen die finale Prüfungsbewertung nur auf einer individuellen Leistung beruhen darf, sind auch die Einzelbeiträge nachvollziehbar. Dennoch wird auch der Gruppenerfolg gemessen und bewertet. Es ist hierbei zu prüfen, ob direkt die "Lehrleistung" der einzelnen Studierenden und der gesamte Gruppenerfolg nicht Eingang in einen Leistungsnachweis finden kann. Falls dies grundsätzlich nicht möglich ist, besteht die Möglichkeit, die Gruppenleistung gesondert und unabhängig von einem Abschlusszeugnis zu bescheinigen. Diese Art des Nachweises von sozialem Engagement wird zurzeit im schulischen Bereich im Rah-

men einer Zeugnisergänzung genutzt. Eine solche Bescheinigung stellt damit auch einen Motivator dar.

5.4.6 Realisiertes Szenario 3: Unterstützung von Lernergruppen

Um den Erkenntnissen des sozialen Konstruktivismus Rechnung zu tragen und dadurch den Studierenden ein tieferes Verständnis des Stoffes zu ermöglichen, wurden auch Gruppenübungen realisiert. Zunächst werden aufgrund der erfassten Leistungskennzahlen Gruppen zusammengestellt, wobei die Auswahl der Gruppenmitglieder die gesamten zur Lösung der Aufgabe notwendigen Fähigkeiten umfasst. Außerdem wird auf starke Wissensdifferenzen geachtet, damit ein entsprechender Bedarf für eine Kommunikation generiert wird.

Als nächster Schritt werden von CATS Gruppenaufgaben generiert. Diese werden mit einem Schwierigkeitsgrad vorbelegt. Abbildung 5.12 stellt das Prinzip der Grup-

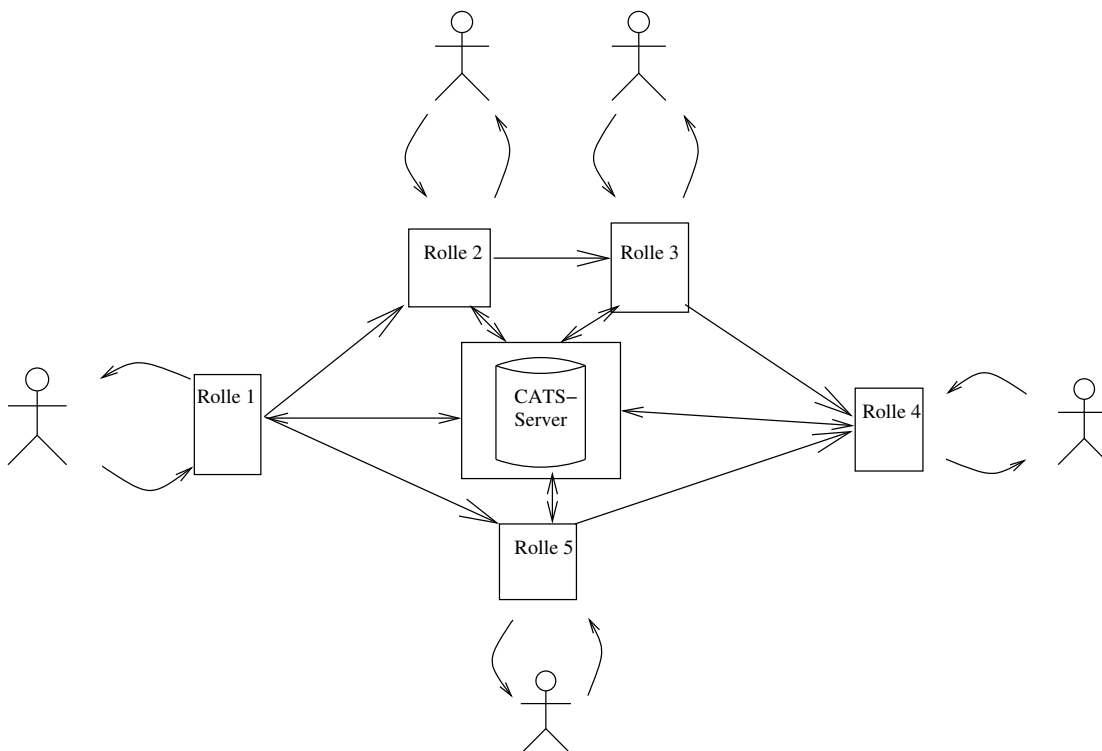


Abbildung 5.12: Gruppenübung mit Kommunikationsunterstützung

penaufgaben näher vor. Der CATS-Server generiert die Aufgaben, verwaltet den Zugriff auf die Aufgaben, überwacht die Leistungsmessung und unterstützt die Grup-

penkommunikation. Hierbei kann man sich die Gruppenaufgaben als eine Art von "Mini-Projekt" vorstellen. Jedem Gruppenmitglied werden bestimmte Rechte zur Beeinflussung des Gruppenobjektes eingeräumt. Neben geschlossenen Lösungen, die eindeutig richtig oder falsch sein können, besteht auch die Möglichkeit, offene Lösungen zu erarbeiten, wie z. B. bei einer simulierten Gerichtsverhandlung. Das Ziel ist die Urteilsfindung unter der Beteiligung der Rollen "Staatsanwalt", "Verteidiger" und "Richter". Alle Beteiligten Gruppenmitglieder arbeiten an dem gleichen Fall, jedoch in unterschiedlichen Rollen.

5.5 Die CATS-Aufgaben-Autorenwerkzeuge

Insbesondere für Studierende der Informatik ist die Programmierung von CATS-Übungsaufgaben in Form von Java-Applets geeignet, im Sinne des Konstruktivismus (s. Abschnitt 2.3.3) vertiefende Kenntnisse in einer bestimmte Wissensdomäne zu erhalten. Bei Anwendung des CATS-Systems in anderen Fachbereichen, wo programmiertechnische Kenntnisse und Erfahrungen nicht vorausgesetzt werden können, würde wahrscheinlich die Akzeptanz von CATS mangels entsprechender Übungsaufgaben bzw. deren aufwendige Erstellung auf Vorbehalte stoßen. Aus diesem Grund wurden mehrere Autorenwerkzeuge für CATS entwickelt, um diese Zielgruppe zu unterstützen. Die beiden Autorenwerkzeuge betreffen Multiple-Choice-Fragen einerseits und Aufgaben in Projektform, wie sie z. B. im Rahmen der Netzplantechnik darstellbar sind, andererseits.

5.5.1 Der Multiple-Choice-Editor

Der Multiple-Choice-Editor wurde weitestgehend vom Projekt WIL-MA (Wireless Interactive Learning -Mannheim) von Nicolai Scheele [SME⁺03] übernommen. Mit diesem Editor kann der Lehrer einfache Multiple-Choice-Fragen generieren. Diese werden in eine XML-Datei geschrieben. Abbildung 5.13 stellt das User-Interface dieses Editors dar.

Hierbei ist für jedes Lernziel eines Faches eine zugeordnete Datei zu erstellen. Den Fragen werden entsprechende Schwierigkeitsgrade zugewiesen, zusätzlich besteht die Möglichkeit, erklärende Texte bei der Anzeige der Lösung einzublenden. Diese sollten ebenfalls didaktisch sinnvoll geschrieben werden. Da CATS eine möglichst feingranulare und den jeweiligen Leistungsstand anpasst, sind entsprechend zahl-

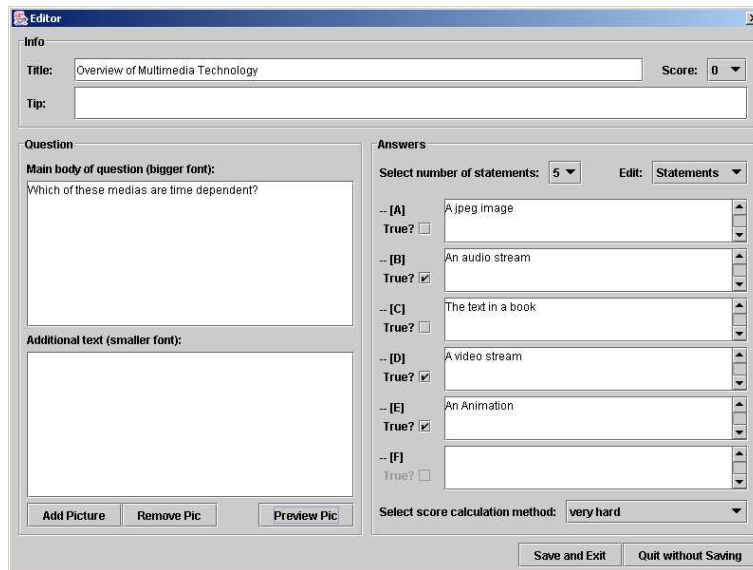


Abbildung 5.13: Der WIL-MA-Multiple Choice Editor

reiche Aufgaben zu erstellen. Um eine Wiederholung bei den Studierenden zu vermeiden, sollten pro Schwierigkeitsgrad mindestens 10 Fragen erstellt werden. Die Präsentation der Multiple-Choice-Fragen ist in Abbildung 5.14 dargestellt. Die Studierenden erhalten die richtigen Lösungen angezeigt sowie ggf. eine ausführliche Erklärung zur Lösung.

5.5.2 Der CATS-Aufgabeneditor für projektbezogene Aufgaben

Ein weiteres Autorenwerkzeug für CATS-Aufgaben beruht auf der Idee von Projekt-Ablaufplänen, die im Rahmen von Lösungswegen eine randomisierte Dynamik aufweisen. Die Gestaltung entsprechender Aufgaben erfolgt mittels eines grafischen Entwicklungswerkzeuges. Hierbei wird von einem informationstechnischen Kenntnisstand der Autoren ausgegangen, wie er einem durchschnittlichen Personal-Computer-Benutzer entspricht. Dies umfasst zumindest den sicheren Umgang mit den Metaphern, wie sie z. B. bei geläufigen Office-Produkten verwendet werden. Die Grundidee des Trainers wurde der bekannten Netzplantechnik entliehen, wobei jedoch das Modell entsprechend modifiziert wurde.

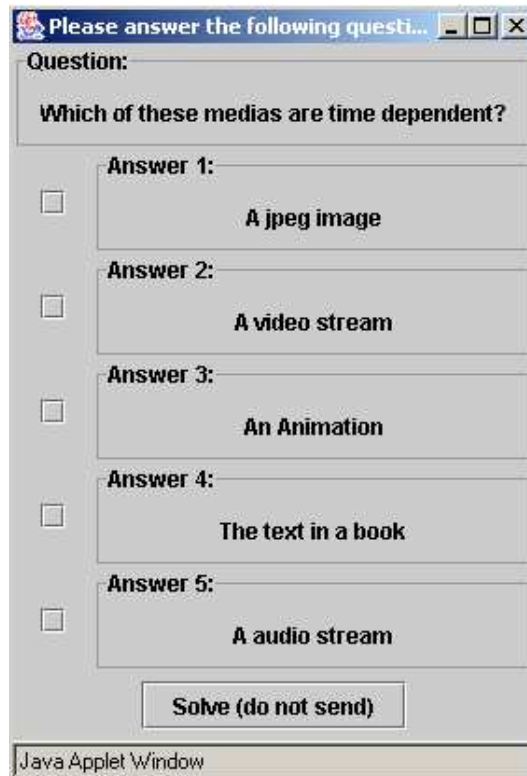


Abbildung 5.14: Das Multiple-Choice Applet

Grundlagen der Netzplantechnik

Die Netzplantechnik (NPT) ermöglicht die Planung, Steuerung und Kontrolle von Projekten. Wenn man die Abwicklung von Aufgaben durch Studierende als "Projekt" auffasst, so ist es nachvollziehbar, dass damit auch entsprechend bekannte Techniken anzuwenden sind. Wesentliche Strukturelemente sind Vorgänge und Ereignisse. Hierbei beanspruchen Vorgänge Ressourcen und Zeit, sie sollten atomar, d. h. nicht weiter zerlegbar sein. Sie verfügen über einen definierten Anfang und ein definiertes Ende. Ereignisse hingegen sind wohldefinierte Zustände im Rahmen des Projektes. [Röd99][S. 6]

In der Netzplantechnik werden im Wesentlichen drei verschiedene Methoden unterschieden:

Critical Path Method (CPM) Bei der Critical Path Method werden Vorgängen Pfeile zugeordnet und Ereignissen Knoten.

Metra Potential Method (MPM) Hier werden Vorgängen Knoten zugeordnet, und

Pfeile stellen Anordnungsbeziehungen dar.

Program Evaluation and Review Technique (PERT) Diese Methode bildet Ereignisse in Knoten ab, und die Pfeile werden als "Geschehnisse zwischen den Ereignissen", ohne sie genauer zu spezifizieren, verstanden.

Im Folgenden wird die Metra Potential Method als Vorlage genommen und die Modifikationen dargestellt.

Modifizierte Netzplantechnik

Die Abbildung 5.15 stellt die logische Abfolge in der Bearbeitung einzelner Objekte grafisch in den aus der Netzplantechnik bekannten Verfahren dar.

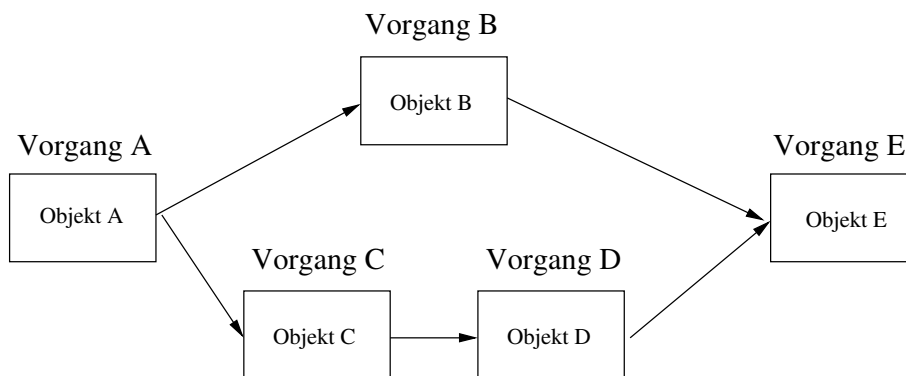


Abbildung 5.15: Modifizierte Netzplantechnik in Objektform

Die verschiedenen Objekte stehen einerseits miteinander in einer logischen Beziehung, andererseits können sie untereinander Nachrichten austauschen und auf die jeweiligen Funktionalitäten zugreifen.

Ein Beispiel soll die Vorgänge illustrieren. Der Lerner soll die Schritte angeben, die notwendig sind um einen Kuchen zu backen. Die Art des Kuchens wird von Stufe zur Stufe schwieriger werden. Es wird zunächst mit einem einfachen Rührkuchen gestartet. Die Anzahl der Personen wird nun durch einen Zufallszahlengenerator festgelegt und ändert sich immer von Aufgabe zu Aufgabe.

Die Erstellung einer Aufgabe mit dem CATS-Aufgabeneditors

Abbildung 5.16 stellt die Oberfläche des Editors dar. Im oberen Bereich befinden sich die Windows-konformen Steuerungselemente des Editors. Diese sind zuständig für

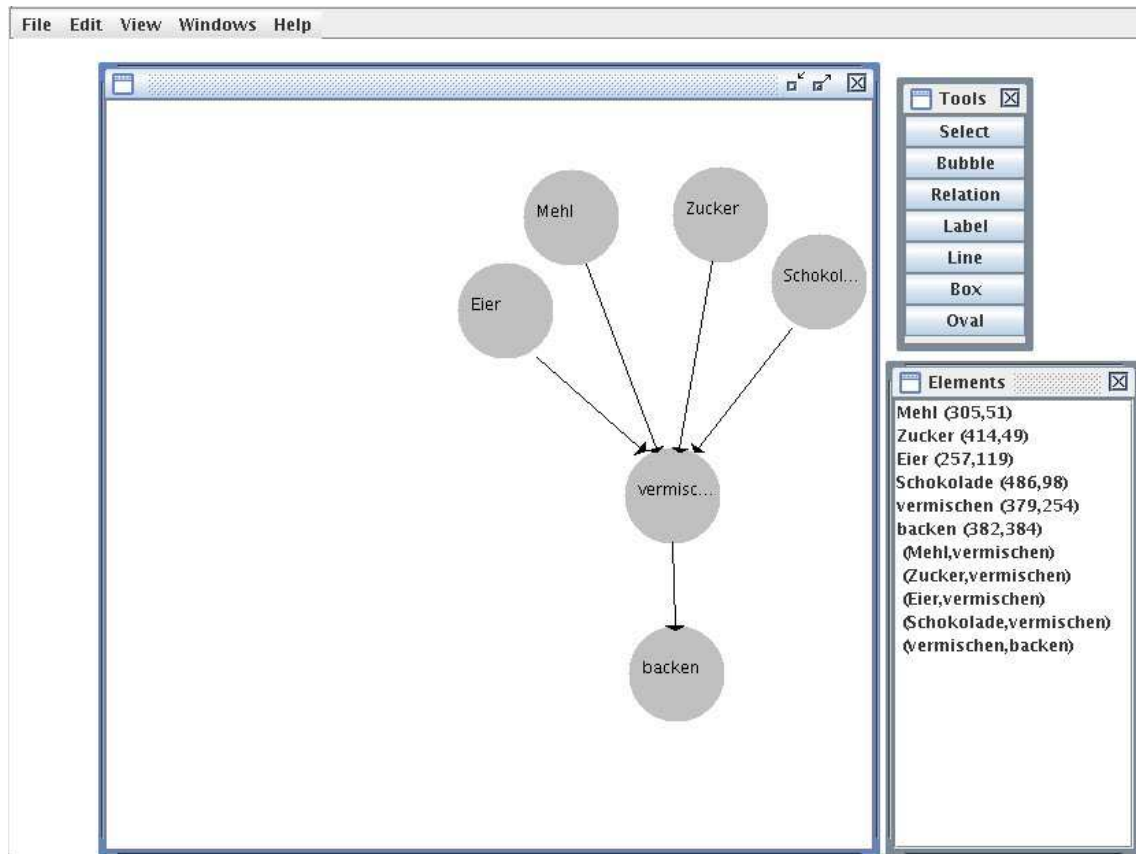


Abbildung 5.16: Die Oberfläche des Aufgabeneditors

das Sichern und Wiederherstellen, die Erzeugung des Übungsapplets, die Einstellungen und schließlich die Hilfe-Funktion.

Am rechten Rand befindet sich die Werkzeugleiste, welche die verschiedenen Elemente enthält, die dem Autor für die Bearbeitung der Aufgabe zur Verfügung stehen. Einerseits sind dies die generischen Objekte, die z. B. nachträglich jeweils mit Bildern geladen werden können, die Relationen, welche die logischen Abhängigkeiten darstellen, und einige grafische Elemente, die der Ausgestaltung des Übungsraums dienen können.

Um einen Backtrainer zu entwickeln braucht der Aufgabenautor nur die einzelnen Schritte mit Hilfe von Blasen und den entsprechenden logischen Relationen einzzeichnen und den einzelnen Blasen, die entsprechende Objekte repräsentieren, Parameter hinzuzufügen.

Sobald er eine entsprechende Anzahl an Backrezepten eingegeben hat, kann die entsprechende Aufgabe den Lernern zur Verfügung stellen.

5 Architektur des Communication and Tutoring System (CATS)

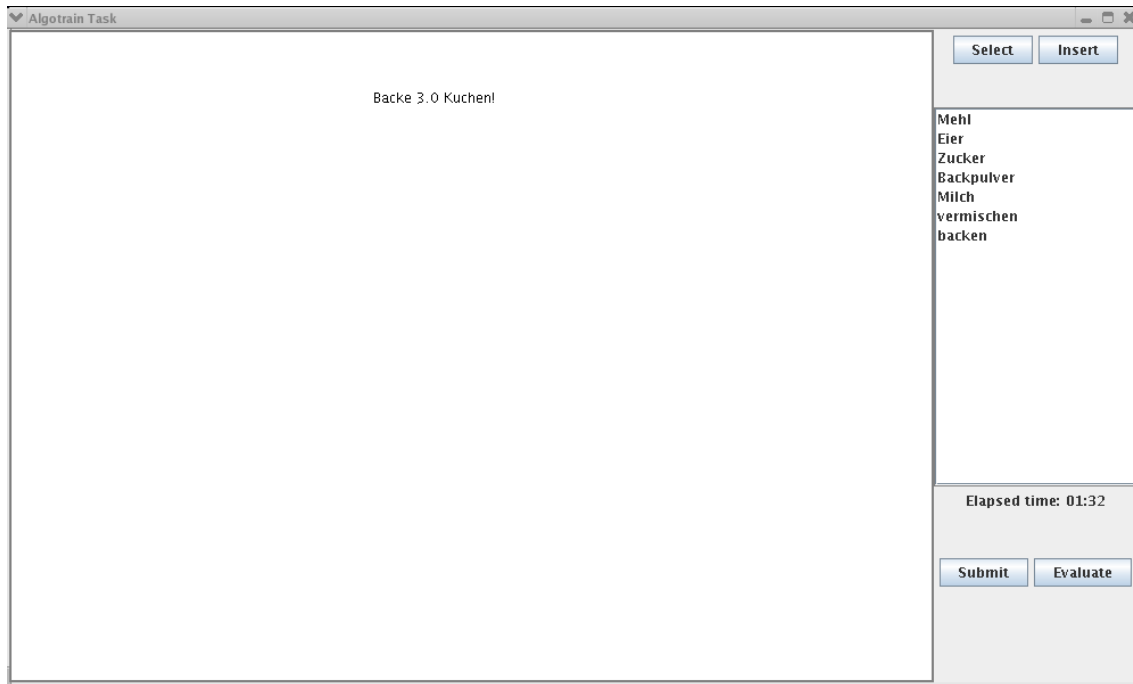


Abbildung 5.17: Der Backtrainer

Abbildung 5.17 zeigt das Bild, welches sich z. B. einem Bäckerlehrling zeigen würde. Er sieht auf der rechten Seite die verfügbaren Objekte und muss diese nun entsprechend der richtigen Lösung anordnen und die entsprechenden Quantitäten berechnen und in die entsprechenden Objekte eintragen.

5.6 Die Client-Server-Architektur

CATS basiert auf einer Client-Server-Architektur. Auf der Server-Seite verwendet CATS eine relationale Datenbank, um die Persistenz der Daten und Gruppenobjekte sicherzustellen. Die Anwendungsschicht ist zweigeteilt. Die Kommunikationsverwaltung wurde durch Server-Anwendungen realisiert, die Übungsapplets werden jedoch auf dem Client ausgeführt. Insgesamt ergibt sich dadurch eine Struktur, die in Abbildung 5.18 dargestellt ist. Der Server ist auf der linken Seite zu sehen, der Client auf der rechten Seite.

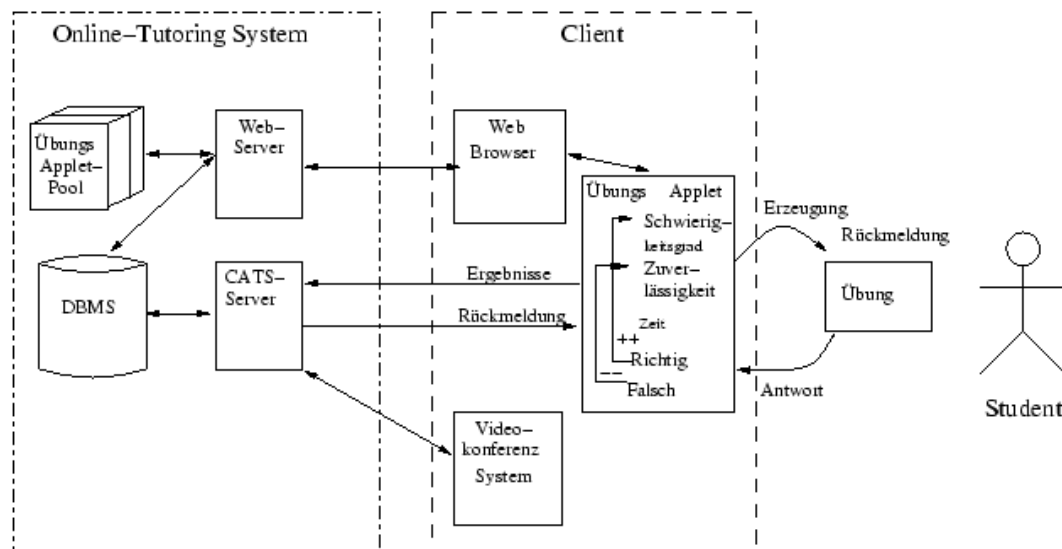


Abbildung 5.18: Die Systemstruktur von CATS

5.7 Der CATS-Server

Der CATS-Server dient der Verwaltung der Studierenden, ihrer Ergebnisse, der Applets sowie der Kommunikationssteuerung. Er verfügt über die Datenbank, die Anwendungsprogramme zur Kommunikationssteuerung, die Programme zur dynamischen Rückmeldung und über die MCU.

5.7.1 Die Datenbankschicht von CATS

Auf der untersten Schicht, der Datenbankschicht, wird von CATS ein relationales Datenbankmanagementsystem (DBMS) verwendet. Die Struktur der Datenbanktabellen ist durch drei Bereiche bestimmt: Die Studierendenverwaltung, die Appletverwaltung sowie die Übungsverwaltung.

Die Studierendenverwaltung

Der Verwaltungsbereich wird durch die Tabelle "students" bestimmt. In dieser Tabelle werden neben den Authentifizierungsdaten und Kontaktdaten auch Informationen über die aktuelle IP-Adresse des Studenten gespeichert. Eine weitere Tabelle namens "groups" dient der Gruppenverwaltung. Diese wird insbesondere für die rollenbasierten Gruppenarbeitsapplets benötigt. Diese Tabelle stellt hierzu die Quelle dar, aus der die Zuordnungen der einzelnen Rollen innerhalb der Gruppenapplets erfolgt.

Die Appletverwaltung

CATS enthält zum einen eine Appletverwaltung, die das Einpflegen neuer Applets und die Übermittlung der Originalapplets erlaubt, zum anderen einen Bereich, der den jeweiligen Zustand der Gruppenobjekte persistent hält. Hierzu werden die serialisierten Objekte in einem binären Feld abgespeichert. Diese Technik kommt insbesondere bei der rollenbezogenen Gruppenarbeit zum Einsatz.

Der Übungsbereich

Der Übungsbereich wird in der Datenbank hauptsächlich durch die Tabelle "results" repräsentiert. In dieser Tabelle werden sämtliche Ergebnisse aller Studierenden festgehalten, die an den CATS-Server übermittelt wurden. Weiterhin werden Operationen zur Auswertung wie z. B. die aktuelle Ranglistenberechnung auf dieser Tabelle durchgeführt.

Weitere Tabellen

Außerdem existieren noch einige Hilfstabellen, die insbesondere der Evaluierung und Fehlerkorrektur des Systems dienen. Die Tabelle "logins" z. B. hält alle Verwaltungsaktivitäten der Nutzer fest.

5.7.2 Die Anwendungsschicht

Die Anwendungsschicht übernimmt die Steuerung des Gesamtsystems. Hierbei sind insbesondere folgende Funktionen zu erfüllen:

- Authentifizierung der Teilnehmer
- Bereitstellung der Übungsaufgaben
- Berechnung und Rückmeldung der Ranglisten
- Kommunikationsunterstützung

Zur Anwendung kommen zum einen Java-basierte Server, die per JDBC direkt auf die Datenbank zugreifen. Diese dienen der Ergebniseingabe und der Rückmeldung der Ranglistenpositionen. Eine andere Anwendung ist der Kommunikations-Server, der die jeweiligen Kommunikationspartner für die gerade bearbeitete Aufgabe ermittelt.

Zudem werden einige Verwaltungsaktivitäten durch ein WWW-Interface gesteuert. Dieses Interface wurde mit der Skriptsprache PHP realisiert. PHP ist eine Programmiersprache, die durch den Webserver interpretiert wird. Sie erlaubt sowohl eine Datenbankbindung wie auch eine einfache Bearbeitung von Web-Eingaben.

5.8 Der CATS-Client

Der CATS-Client ermöglicht dem Lerner die Administration, die Bearbeitung der Übungsaufgaben und die Verbindungsaufnahme zu anderen Lernern.

Die Abbildung 5.19 stellt die Benutzerschnittstelle des CATS-Systems dar. Zur leichten Navigation befindet sich auf der linken Seite eine Navigationsleiste, die dynamisch vom CATS-Server erstellt wird. Sobald ein neues Applet verfügbar wird, erscheint es bei dem Benutzer in der Navigation in dem jeweiligen Themengebiet. Der Benutzer kann durch Anklicken die entsprechende Übungsaufgabe auswählen. Diese erscheint sofort im mittleren Bereich. Zeitgleich werden die zu dieser Aufgabe zur Verfügung stehenden Studierenden im rechten Bereich angezeigt. Sie erscheinen dort mit ihrer Kennung und in der Reihenfolge ihrer zu dieser Aufgabe maximal erzielten Punktzahl. Die Studierenden haben nun, ebenfalls durch Anklicken, die Möglichkeit, sich direkt mit ihren Kommilitonen zu verbinden. Die Anwesenheitssteuerung erfolgt durch das Instant Messaging Tool, welches sich im unteren rechten

5 Architektur des Communication and Tutoring System (CATS)

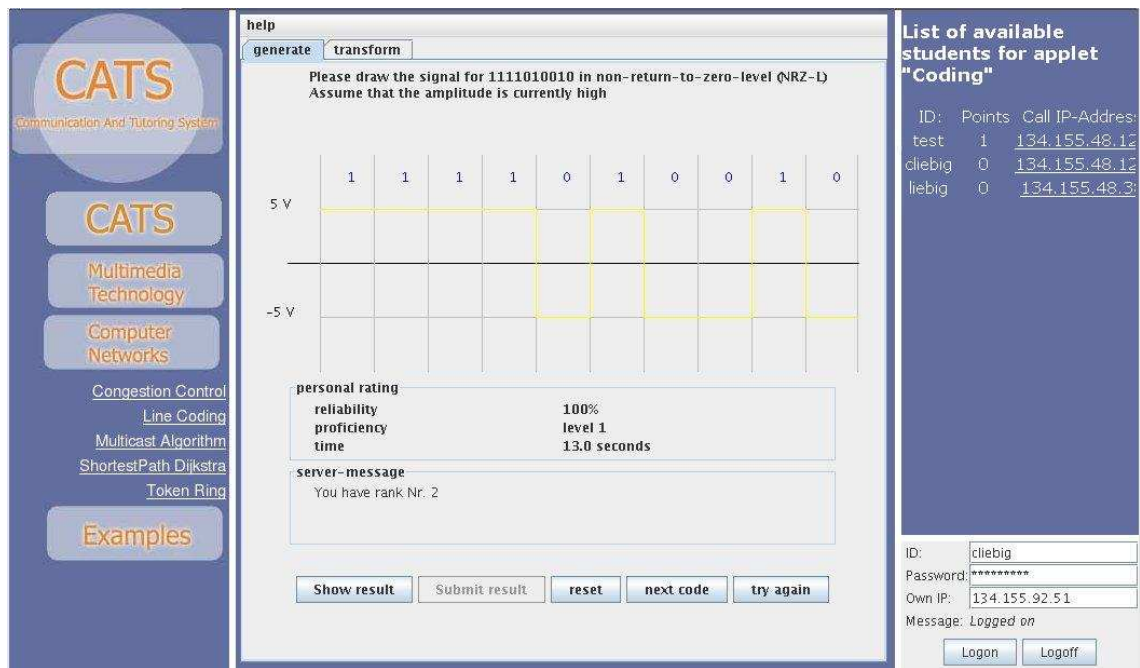


Abbildung 5.19: Die Benutzerschnittstelle von CATS

Bereich befindet. Nach Eingabe ihrer Authentifizierungsdaten haben die Studierenden die Möglichkeit, ihre Bereitschaft zur Beantwortung von Anfragen zu signalisieren. Dies erfolgt durch den Knopf "logon", ebenfalls können sie, falls die Bereitschaft nicht mehr vorhanden ist, dies mit dem Knopf "logoff" dem System signalisieren. Ihre Kennung wird dann nicht mehr angezeigt.

5.9 Integrationsmöglichkeiten von CATS in bestehende Lernplattformen

Die immer weitere Verbreitung von integrierten Lernplattformen macht es erforderlich, durch ein entsprechendes Integrationskonzept eine weitgehende Integration des CATS-Systems zu ermöglichen. Dies setzt neben einer Möglichkeit des Datenimports und Exports beim CATS-System eine entsprechende Möglichkeit bei den jeweiligen Zielpattformen voraus. In aller Regel wird durch die meisten Hersteller ein so genanntes Application Programming Interface (API) zur Verfügung gestellt.

5.9.1 Das Integrationsmodell von CATS

Um die Integration zu erleichtern, existiert für das CATS-System ein Integrationsmodell. Dies geht von der oben vorgestellten Schichtenarchitektur aus und bietet entsprechende Integrationsmöglichkeiten pro Schicht an.

Abbildung 5.20 stellt das CATS-Integrationsmodell schematisch dar. Auf der unter-

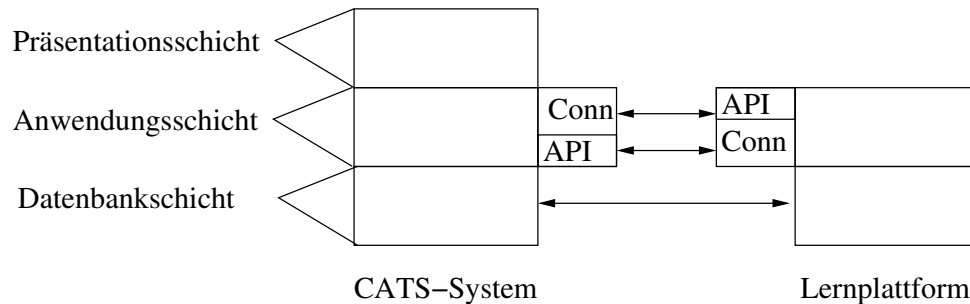


Abbildung 5.20: Das CATS-Integrationsmodell

ten Schicht, der Datenbankschicht, besteht die Möglichkeit der Integration durch Benutzung des gleichen Tabellenraums. Sehr viele Lernplattformen benutzen ebenfalls auf der untersten Ebene eine relationale Datenbank. In bestimmten Fällen kann die gleiche Datenbank und der gleiche Benutzer für beide Systeme verwendet werden. Hier sehen beide Systeme die gleichen Tabellen und können ebenfalls auf diese zugreifen. Dies wäre die engste Form der Integration.

Eine weitere Möglichkeit besteht in einer Integration auf der Anwendungsebene. Hierzu ist es allerdings notwendig, dass die Lernplattform eine so genannte API zur Verfügung stellt, um auf die entsprechenden, für die Integration erforderlichen Funktionalitäten zugreifen zu können. Das Programm, welches die API benutzt, kann entweder einen Push-Mechanismus (z. B. Eingabe der CATS-Ergebnisse in die Lernplattform) oder einen Pull-Mechanismus (z. B. Einlesen der Authentifizierungsinformationen in das CATS-System) in Gang setzen. In der Darstellung wurde dieses Programm als "Conn" (für Connection Program) bezeichnet. Diese Art der Integration hat den Vorteil, dass alle Zugriffe durch die Anwendungsschicht kontrolliert werden können und sich dadurch die Abläufe beider Systeme besser integrieren lassen, während bei der Integration auf der untersten Schicht die Konsistenzerhaltung ein Hauptproblem darstellt, denn es sind die Abhängigkeiten innerhalb beider Systeme zu beachten.

Eine weitere Integrationsmöglichkeit bestünde darin, dass die erstellten Übungsaufgaben den Studierenden durch die Lernplattform zur Verfügung gestellt werden. Dies würde dem vorgestellten Lern-Szenario 1 entsprechen, wobei die Lernplattform dann lediglich als Distributionsmedium dient.

5.9.2 Die Integrationsmöglichkeiten

Es wurden zwei Lernplattformen näher auf die Möglichkeiten einer Integration hin untersucht. Es handelt sich hierbei um das Open-Source-Projekt .LRN und um das kommerzielle Blackboard Learning System.

Integrationsmöglichkeit von CATS in .LRN

Das bereits in Abschnitt 3.8.6 vorgestellte System .LRN bietet mehrere Möglichkeiten der Integration an. Es verwendet selbst auf unterster Schicht eine relationale Datenbank, die die gleiche ist wie sie bei CATS (postgresql).

Eine weitere Möglichkeit besteht in der Integration auf der Anwendungsebene. Hierbei muss jedoch, falls ein Modul in .LRN erstellt wird, dieses zunächst in der von .LRN verwendeten Open Architecture Community System (openACS), das wiederum in TCL programmiert wurde, erstellt und durch eine ODBC-Verbindung mit der CATS-Datenbank verbunden werden.

Die dritte Integrationsmöglichkeit bestünde, wie oben dargestellt, in der Form, dass die Übungsaufgaben, da sie auch für sich allein funktionsfähig sind, als Lernmodule in die Lernplattform importiert werden. Hier bietet es sich an, mithilfe eines SCORM-Editors (z. B. Reload [Mil05]) pro Applet noch entsprechende Metainformationen hinzuzufügen und dies dann als SCORM-Objekt in die Lernplattform zu importieren. Die .LRN-Lernplattform unterstützt zum Zeitpunkt der Erstellung dieser Arbeit den SCORM-Standard 1.2.

Integrationsmöglichkeit von CATS in das Blackboard Learning System

Da das Blackboard Learning System, wie in Abschnitt 3.8.4 vorgestellt, ein kommerzielles System ist, beschränken sich die Integrationsmöglichkeiten auf solche, welche durch den Hersteller offen gelegt und dokumentiert sind. Im Falle von Blackboard bestünde die Möglichkeit, durch eine definierte Java-API (im Rahmen des Building Block-Programms) auf die Funktionalitäten der Lernplattform zuzugreifen.

5.9 Integrationsmöglichkeiten von CATS in bestehende Lernplattformen

Dies würde die Integration von CATS auf der Ebene der Anwendung auf Server-Seite ermöglichen. Auch hier wäre eine weitere Möglichkeit der Integration der Import von Übungsaufgaben als SCORM-Learning-Objects.

6 Die Implementierung des Communication and Tutoring Systems (CATS)

Wie bereits im vorangegangenen Kapitel dargelegt, ist die Architektur des CATS-Systems Client-Server basiert. Entsprechend erfolgte auch die Implementierung des Systems. Innerhalb der Server-Architektur ist das System auf Schichten aufgebaut, um die Integration in andere Systeme, insbesondere Lernplattformen, zu erleichtern.

6.1 Die Implementierung des CATS-Servers

Zur Realisierung des CATS-Servers wurde ein Linux-System verwendet. Das System wurde mit zwei verbreiteten Linux-Distributionen getestet: Zum einen mit einer SuSE 8.2-Distribution und zum anderen mit einer RedHat Version 8.0. Beide Distributionen verfügen über die gesamte Software, die für den Betrieb und für die Programmierung des Systems notwendig sind. Beide Distributionen enthalten insbesondere einen Web-Server (Apache), ein relationales, SQL-kompatibles Datenbankmanagementsystem (PostgreSQL), eine Java-Distribution (Suns Java 1.4.3) sowie alle weiteren notwendigen Hilfsprogramme.

6.1.1 Das CATS-Datenbanksystem

Für CATS wurde das relationale und SQL-kompatible Datenbankmanagementsystem PostgreSQL verwendet. Hierfür gabe es folgende Gründe:

1. PostgreSQL steht kostenfrei zur Verfügung. Sie unterliegt einer freien Lizenz der University of California [Mom01] [S. 264 f.].

2. Das System verwendet die Datenbankabfragesprache Structured Query Language (SQL) und erlaubt damit eine leichte Portierung der entwickelten Anwendungen auf andere, insbesondere kommerzielle Datenbanksysteme (wie z. B. Oracle), welche ebenfalls diese Abfragesprache einsetzen.
3. Das Datenbanksystem verfügt über eine ausreichende Performanz und Reife, so dass ein stabiler und effizienter Betrieb gewährleistet werden kann.
4. Die Datenbank unterstützt die Verarbeitung binärer Daten, es eignet sich aus diesem Grund für die Verwaltung von serialisierten Java-Objekten, die insbesondere im Rahmen der Gruppenarbeitsmodule verwendet werden.

Der CATS-Tabellenaufbau

Abbildung 6.1 stellt die von CATS verwendeten Tabellen in einem Entity-Relationship-Diagramm [Che76] symbolisch dar. In der Grundversion des CATS-Systems, die insbesondere die Szenarien 1 und 2 (s. Abschnitte 5.2 und 5.3) unterstützt, sind die Tabellen *Students*, *Applets* und *Results* notwendig.

Die Tabelle *Applets* dient der Applet-Verwaltung. Sie enthält einerseits die Information, wo sich das Applet befindet (URL), die Einordnung in das jeweilige fachliche Kapitel und die Soll-Vorgaben der Werte Proficiency, Reliability und Time, die zur Abschlussprüfung erforderlich sind. Über die Applet-ID als primären Schlüssel erfolgt die entsprechende Zuordnung in anderen Tabellen.

Die Tabelle *Students* dient zur Verwaltung der Studierenden. Sie enthält die gesamten Registrationsdaten der Studierenden. Hierzu zählen neben dem vollständigen Namen auch die E-Mail-Adresse, die im Rahmen der Aktivierung der Registrierung überprüft wird, und die Kommunikationsinformationen. Diese sind für die H.323-konforme Kommunikation die aktuell gültige IP-Adresse (die über das Anmeldeapplet durch eine Systemanfrage automatisiert ermittelt wird) oder für die SIP-Telefonie die SIP-Adresse.

Die Tabelle *Results* enthält neben der Applet-ID und den in Abschnitt 5.2.1 vorgestellten KPIs auch die Authentifizierungsdaten der Studierenden und einen Zeitstempel des jeweiligen Eintrags.

Für die Gruppenarbeitsfunktionalitäten, die im Szenario 3 beschrieben wurden (siehe hierzu Abschnitt 5.4.6), kommen noch weitere Tabellen hinzu: *Groups*, *Workobject* und *Patternobject*. Durch die Matching-Anwendung werden die Studierenden nach den in Abschnitt 5.4.1 festgelegten Regeln in entsprechende Lerngruppen eingeteilt.

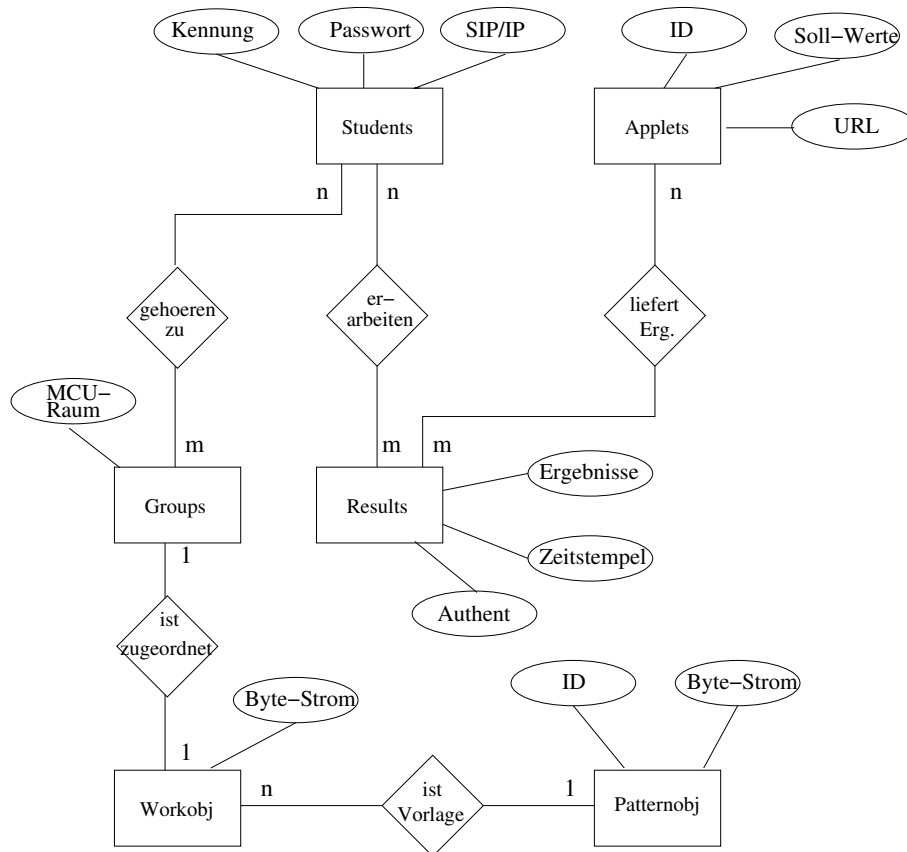


Abbildung 6.1: Das Entity-Relationship-Diagramm der CATS-Datenbank

Diese Zuordnung erfolgt zunächst in der Tabelle groups. Entsprechend lässt die Tabelle sowohl die Zuordnung in Arbeitsgruppen (Metagruppen) wie auch Unterarbeitsgruppen bzw. Rollen zu. Um die für die Gruppenkommunikation erforderlichen Informationen bereitzustellen, werden sowohl die MCU-Räume wie auch die Adressen der jeweiligen zuständigen MCUs pro Eintrag festgehalten.

Nachdem die Gruppenzuordnung durchgeführt wurde, wird auf der Applikations-ebene aus der Tabelle Patternobj ein serialisiertes Java-Gruppenobjekt ausgewählt und entsprechend dem Proficiency-Level der Gruppe parametrisiert. Während der Abwicklung der Gruppenaufgabe werden dann die jeweiligen Zwischenstände in der Tabelle Workobject persistent gehalten.

Beziehungen zwischen den Tabellen

Die Beziehungen zwischen den einzelnen Tabellen sind in Abbildung 6.1 dargestellt:

Die Studierenden können keiner, einer oder mehreren Gruppen angehören. Zusammen mit den jeweiligen Applets können pro Applet beliebig viele Ergebnisse generiert werden, die dann in der Tabelle Results festgehalten werden. Diese Tabelle dient der späteren Ergebnisauswertung und damit auch der Gruppenzusammenstellung. Bezüglich der Gruppenarbeit ist zu beachten, dass jeweils einer Gruppe auch nur genau ein Exemplar des Gruppenarbeitsobjektes aus *Workobject* zugeordnet wird. Es können aber beliebige Exemplare an Musterobjekten nach der entsprechenden Parametrisierung zu Gruppenarbeitsobjekten werden. Diese müssen dann aber jeweils anderen Gruppen zugeordnet werden. Jeder Eintrag, sowohl in der Tabelle *Patternobj* wie auch *Workobj*, enthält daher ein Attribut "Byte-Strom", welches das serialisierte Java-Gruppenarbeitsobjekt abspeichert und damit persistent hält.

6.1.2 Die CATS-Anwendungsschicht

Abbildung 6.2 stellt die einzelnen CATS-Anwendungen schematisch dar. Die Anwendungsprogramme übernehmen die Steuerung der Tabelleneinträge und die Kommunikation mit den Client-Anwendungen. Diese sind insbesondere die Navigation, die Awareness-Funktion, die Ergebnis-Eingabe und die Verarbeitung der übermittelten Ergebnisse. Soweit CATS als Gruppenaufgabenserver eingesetzt wird, erfolgt die Verwaltung und Steuerung der Aufgaben ebenfalls durch eine eigene Anwendung.

Der CATS Anmelde-Server

Der CATS Anmelde-Server ist eine in Java-geschriebene Applikation, die auf Port 3250 auf einen Verbindungsaufbau seitens des Clients wartet. Der Anmelde-Server stellt folgende Grundfunktionen zur Verfügung:

Logon Die Studierenden signalisieren damit ihre Bereitschaft, für Anfragen anderer Studierender verfügbar zu sein. Im Rahmen dieser Funktion wird die aktuelle IP-Adresse der Studierenden im entsprechenden Tabelleneintrag gesetzt.

Logoff Diese Funktion löscht den IP-Adress-Eintrag in der Tabelle. Dadurch signalisiert die/der Studierende, dass sie/er nicht mehr bereit ist, etwaige Anfragen zu beantworten.

GetStatus Durch diese Funktion wird der aktuelle Status des Studierenden durch eine Datenbankabfrage ermittelt und dem Client übermittelt.

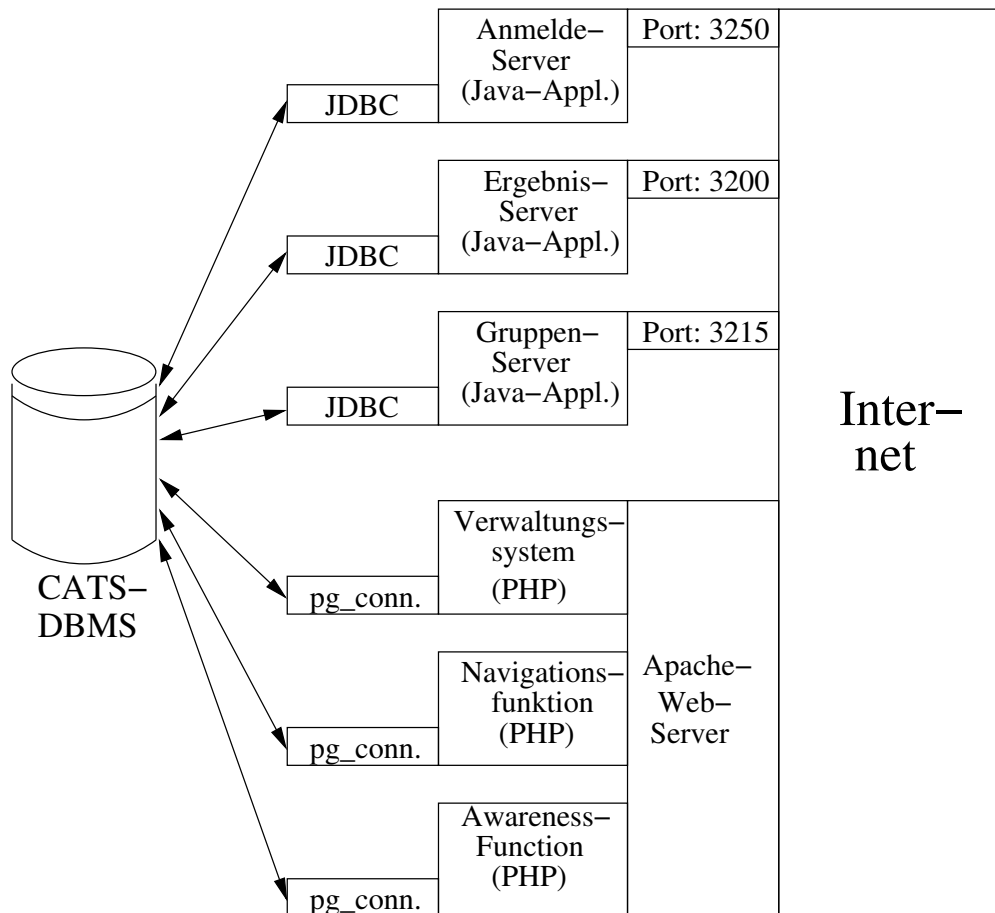


Abbildung 6.2: Die CATS-Anwendungen

Call Durch diese Funktion wird der für ein bestimmtes Applet zurzeit verfügbare beste/bester Studierende ermittelt und dem Client übermittelt.

Der CATS-Ergebnis-Server

Der CATS-Ergebnis-Server ist ebenfalls eine Java-Applikation, die auf Port 3200 auf eine eingehende Verbindung wartet. Es werden die Authentifizierungsdaten empfangen und zusammen mit den entsprechenden Leistungswerten sowie der Applet-ID zuzüglich eines Zeitstempels in die Tabelle "Results" eingetragen.

Da die Studierenden auch eine pädagogisch sinnvolle Rückmeldung erhalten sollen, besteht die Möglichkeit, eine entsprechende Mitteilung zurückzusenden. In der jetzigen Version wird die aktuelle Rangposition des Studierenden für die jeweils aktuell bearbeitete Aufgabe ermittelt und dem Client übermittelt.

Gruppenaufgabenserver

Der Gruppenaufgabenserver wurde als Java-Applikation realisiert. Er dient zur Abwicklung kompletter, rollenbasierter Gruppenaufgaben. Der Server verwaltet zum einen die Schemata der einzelnen unterschiedlichen Aufgaben, parametrisiert die Aufgaben mit dem entsprechenden Gruppen-Proficiency-Level und überwacht den ordnungsgemäßen Ablauf der Aufgabenbearbeitung. Hierzu wurden insbesondere folgende Methoden prototypisch implementiert:

new Mit dieser Methode wird das Gruppenarbeitsobjekt initialisiert und parametrisiert. Hierbei wird zunächst durch Introspektion der Klassenname des eingebetteten Objektes ermittelt. Dieses Verfahren wurde gewählt, um zur Laufzeit des Systems neue Gruppenaufgaben realisieren zu können, ohne dass es zu einer Störung des Betriebes kommt. Es erfolgt die Erzeugung eines entsprechenden Objektes und schließlich die Vorbelegung mit einem Schwierigkeitsgrad und einem Benutzer, der Zugriff auf die Verwaltungsfunktionalität des internen Objektes erhält. Dieser kann nun ggf. das Objekt weiter parametrisieren.

store Diese Methode dient dazu, ein über Objektserialisierung übertragenes Aufgabenobjekt persistent in CATS zu halten. Hierbei wird über eine fortlaufende Objekt-ID sichergestellt, dass es zu keinen Inkonsistenzen kommt. Das Gruppenarbeitsobjekt wird dann in der Tabelle Workobjects im entsprechenden Eintrag als Byte-Strom abgespeichert.

restore Die Restore-Methode dient dazu, ein abgespeichertes Gruppenarbeitsobjekt wieder dem jeweiligen Übungsapplet zur Verfügung zu stellen. Hierzu wird zunächst anhand der Objekt-ID das dazugehörige Objekt in der Tabelle Workobject ermittelt und dieses dann anschließend an das Applet übertragen.

delete Diese Methode dient der Bereinigung von alten, nicht mehr verwendeten Objekten.

Das CATS-Verwaltungssystem

CATS verfügt über ein Verwaltungssystem, das Exercise Administration and Controlling System. Es handelt sich hierbei um eine Anwendung, die in der Skriptsprache PHP entwickelt wurde. Dieses System stellt ein Web-Front-End bereit und

erlaubt eine Abfrage der verschiedenen Tabellen und die Administration des Gesamtsystems. Ein Auszug des Verwaltungssystems ist in Abbildung 6.3 dargestellt. Neben einzelnen Abfragen sind hierdurch auch statistische Auswertungen möglich.

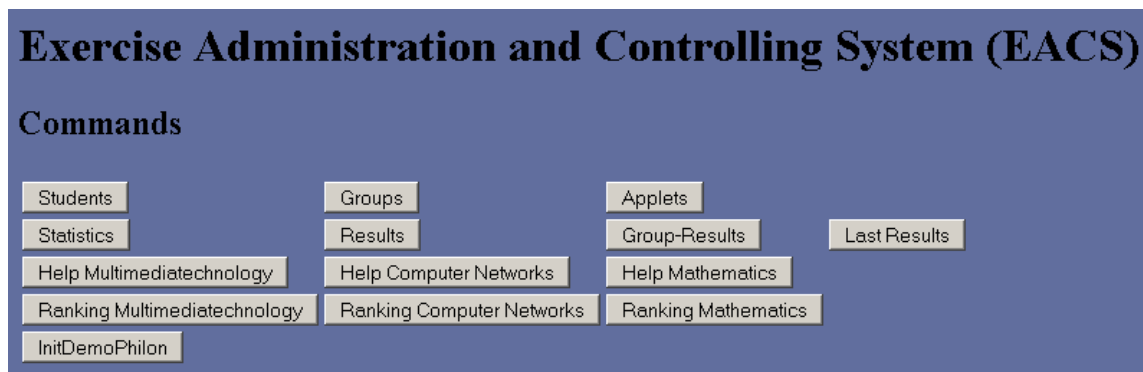


Abbildung 6.3: Das CATS-Verwaltungssystem

Diese werden pro eingetragenes Applet durchgeführt. Die Abbildung 6.4 gibt hierüber einen Eindruck. In dieser Auswertung werden die Mittelwerte der KPIs (s. Abschnitt 5.2.1) pro Aufgabenapplet dargestellt.

Die Navigations-Unterstützung

Die CATS-Navigationsunterstützung generiert für die CATS-Benutzer eine Übersicht der zur Verfügung stehenden Übungsaufgaben, nach dem jeweiligen Fachgebiet geordnet, wie sie schematisch in Abbildung 5.19 auf der linken Seite dargestellt wird. Abbildung 6.5 stellt die Navigationsleiste dar. Die vorhandenen CATS-Übungsapplets werden automatisiert entsprechend dem jeweiligen Themenbereich eingeordnet. Hierzu wird durch das PHP-Skript die Tabelle *Applets* ausgewertet und die entsprechende Navigationsleiste erzeugt. Die einzelnen Aufgaben sind entsprechend durch URLs verlinkt, und die Studierenden können sich nun die ausgewählte Aufgabe anzeigen lassen.

6.2 Die Implementierung der CATS-Clienten

6.2.1 Grundlagen des Usability-Design

Einer der wesentlichen Erfolgsfaktoren für Endbenutzer-Software ist das Usability Design. Hierunter versteht man nach ISO 9241 die Gebrauchstauglichkeit eines Pro-

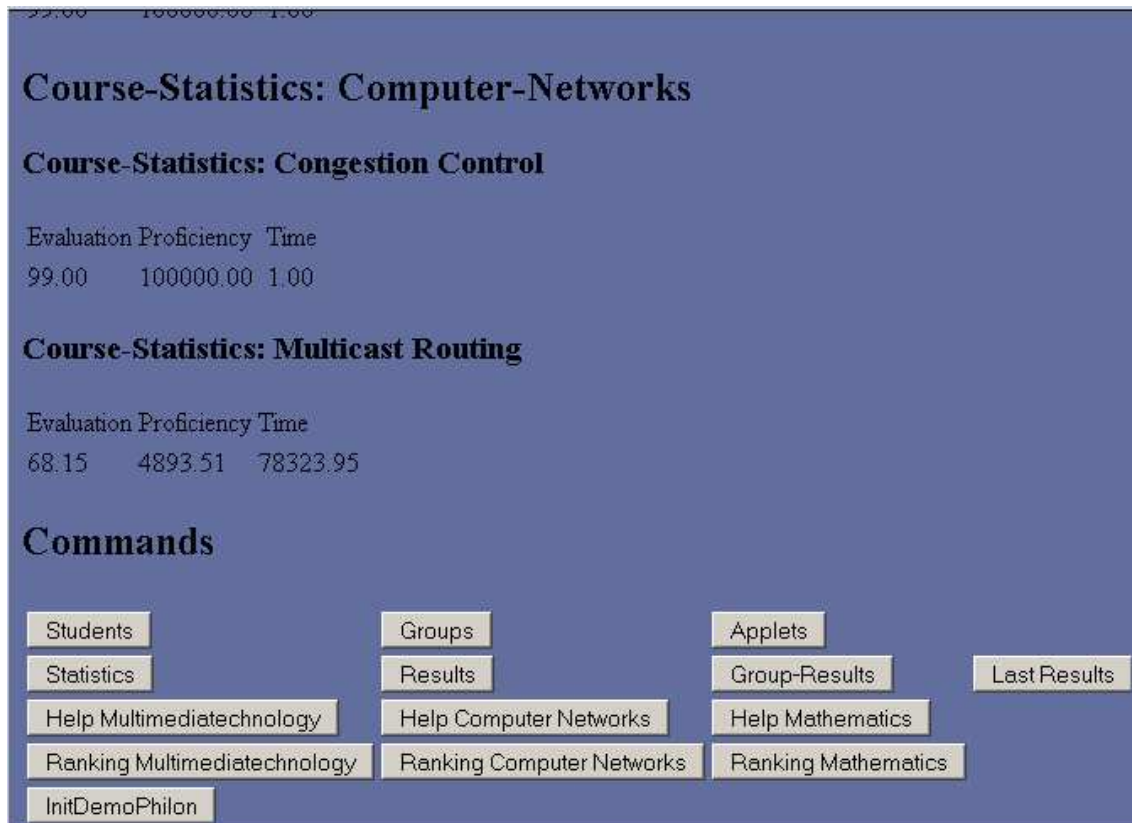


Abbildung 6.4: Statistische Auswertungen durch das EACS

duktes. Diese wird durch die drei maßgebenden Faktoren Effizienz, Effektivität und Zufriedenheit bestimmt.

Während man bei konventionellen Computer-Based-Trainings (CBT) nach der Auswahl der zugrunde liegenden Lerntheorie in enger Verbindung mit dem Instruktionsdesign auch Rückschlüsse auf die Benuteroberfläche ziehen konnte (beispielsweise erfordern behavioristische Vokabel-Lern-Programme ein weniger komplexes Benutzer-Interface als etwa konstruktivistische Ansätze), stellt sich die Problemstellung bei kooperativen Lernumgebungen wie bei CATS wesentlich komplizierter dar. Innerhalb des CATS-Systems gibt es mehrere Ebenen, die unterschiedliche Anforderungen an das Usability Design stellen:

Administrative Ebene (aus der Lernerperspektive) Diese Ebene soll die Lerner über wesentliche Eigenschaften des Systems informieren, ihnen eine intuitive Anleitung bieten und ihnen die formellen Schritte (wie z. B. Registration etc.)

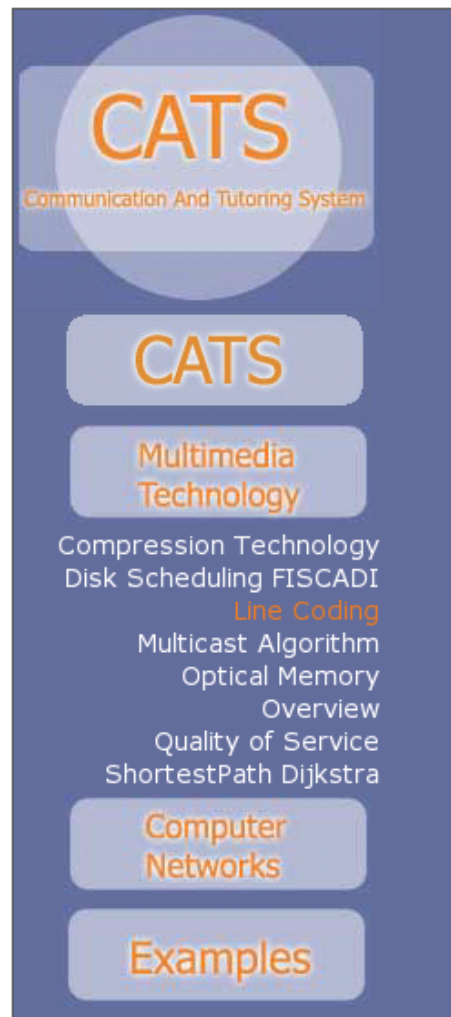


Abbildung 6.5: Die CATS-Navigationsleiste

erleichtern. Um dies zu unterstützen, existiert eine Kurzbeschreibung, welche die Ziele des Systems, die pädagogische Motivation und die Gesamtarchitektur darstellt. Durch diese Offenheit soll entsprechendes Vertrauen gewonnen werden. Um einen schnellen und direkten Start zu ermöglichen, sind entsprechende Verknüpfungen zur Registrierung sowie zum Startpunkt des Übungssystems realisiert worden.

Navigationsbereich Da, wie bereits dargestellt, CATS nicht nur für eine bestimmte Übungsaufgabe konzipiert wurde, sondern zumindest einen gesamten Studiengang mit sehr unterschiedlichen Fächern unterstützen soll, werden an die Navigation erhöhte Anforderungen gestellt. Die einzelnen Übungsaufgaben

sollten zum Beispiel den jeweiligen Themenbereichen zugeordnet werden und in Form eines didaktisch fundierten Curriculums präsentiert werden. Hierbei kann es vorkommen, dass einzelne Übungsaufgaben in mehreren verschiedenen Fächern verwendet werden. Die schlichte Auflistung aller verfügbaren Aufgaben reicht daher nicht aus. Vielmehr wurde die Navigation in einer zweistufigen Gliederung (Fach und Aufgabe) gegliedert. Dies wurde dadurch erreicht, dass in den einzelnen Datensätze in der Tabelle Zusatzinformationen eingetragen wurden, die die Zugehörigkeit der Übungsaufgaben zu den unterschiedlichen Themengebieten beschreiben.

Übungsbereich Der Übungsbereich stellt das Zentrum der Interaktion der Lernenden mit dem System dar. Daher wurde darauf geachtet, dass dieser Bereich sowohl von der Positionierung wie auch von der Größe den optischen Mittelpunkt auf dem CATS-Arbeitsbereich bildet.

Kommunikationsbereich Der Kommunikationsbereich stellt das Besondere dieses Systems dar. Er umfasst aus Benutzersicht heraus zwei wesentliche Elemente: Die eigentliche Kommunikationssteuerung, dies bedeutet insbesondere die zielgerichtete Kontaktaufnahme zu anderen Studierenden, und die Mitteilung der eigenen Kommunikationsbereitschaft. Der andere Bereich ist eine Awareness-Funktion: Den Lernenden soll die gleichzeitige Anwesenheit von weiteren Mitlernern bewusst werden und sie dadurch ermutigen, mit diesen auch Kontakt aufzunehmen. Hierbei dürfen jedoch von der Awarenessfunktion aus keine Störungen des eigentlichen Lernvorgangs ausgehen.

Diese Anforderungen wurden in einem Gesamtkonzept vereint. Bisherige CSCL und CSCW-Systeme vereinheitlichen zwar unter einer Plattform sowohl die Lern- bzw. Arbeitsumgebungen mit Kommunikationsdiensten, diese Funktionen sind jedoch in eigenen Applikationen realisiert und damit voneinander getrennt. Im Usability-Design von CATS wurde jedoch ein neuer Weg beschritten. Die Kommunikationsfunktionalitäten werden direkt mit den Übungsaufgaben verbunden. Gerade bei Gruppenübungen wird die gesamte Kommunikation direkt über das Applet gesteuert. Dieser moderne Ansatz wird als objektzentrierte Kommunikation (s. [GVCM03]) bezeichnet und wird voraussichtlich auch in zukünftigen CSCW-Systemen eingesetzt werden.

6.2.2 Design des Benutzerinterfaces

Auch die Implementierung des Benutzerinterfaces erfolgte nach dem Konzept der objektzentrierten Navigation [GVCM03]. Bisher trennten Kooperationsumgebungen einzelne Funktionalitäten von den zu bearbeitenden Objekten. So war bisher zum Beispiel der E-Mail-Verkehr zu einem Dokument von diesem Dokument getrennt, oftmals wurden auch zwei verschiedene Programme verwendet (z. B. Winword und Pmail). Die Folge ist dann ein Bruch im Bedienparadigma, der durch die Umschaltung zu der anderen Software entsteht. Dies führt zu Ineffizienzen, da die jeweiligen Informationsströme nicht zielgerichtet sind und von den Benutzern jeweils gesondert bearbeitet (in aller Regel besonders gespeichert) werden müssen. Das Konzept der objektzentrierten Navigation geht hier einen anderen Weg: Im Mittelpunkt steht das Objekt, welches gerade gemeinsam bearbeitet werden soll; im Rahmen des CATS-Konzeptes zum Beispiel ein konkreter juristischer Fall, der von den Gruppenmitgliedern "Ankläger", "Verteidiger", "Richter" und "Reviewer" bearbeitet wird. Innerhalb dieses Objektes findet nun die gesamte Kommunikation statt. So arbeiten die jeweiligen Gruppen zum Beispiel an einem gemeinsamen Schriftsatz, die entsprechenden Veränderungen können dann zusätzlich noch in einer Historie festgehalten werden. Dieses Konzept äußert sich auch in der grafischen Darstellung.

6.2.3 Die CATS-Aufgabenmodule

Die CATS-Aufgaben selbst werden als Java-Applets realisiert. Hierbei sind die Entwickler grundsätzlich weitgehend frei in der Verwendung der Funktionalitäten, es sind jedoch bestimmte sicherheitstechnische Restriktionen zu beachten: Im Gegensatz zu Java-Applikationen sind Ressourcen (z. B. Festplatten, Mikrofon-Eingänge, etc.) durch das Java Runtime Environment besonders geschützt. Entweder ist von der Verwendung derartiger Methoden abzusehen, oder das Applet ist entsprechend zu signieren. Es empfiehlt sich jedoch, im Sinne einer gelungenen Gebrauchstauglichkeit Designvorgaben an die Erstellung entsprechender Übungsaufgaben festzulegen. So kann die Anordnung von bestimmten Steuerungsfunktionen und auch fest definierten Begrifflichkeiten, die über alle Applets hinweg die gleichen sind, die Bedienung und Handhabung wesentlich erleichtern.

Um die Übermittlung der Ergebnisse zu dem CATS-Server zu standardisieren und für zukünftige Verfahren und Protokolle erweiterbar zu machen, wurde eine Klasse *Results* für alle Übungsaufgaben fest vorgegeben.

6 Die Implementierung des Communication and Tutoring Systems (CATS)

Die Klasse Results selbst ist *serialisierbar* und bringt auch alle notwendigen Methoden mit. Unter Serialisierung versteht man die Fähigkeit, ein Objekt aus dem Hauptspeicher der Anwendung heraus in ein Format zu konvertieren, das es erlaubt diese Objekt in eine Datei, oder wie in diesem Fall, über das Netzwerk zu transportieren. Die Klasse results beinhaltet die in Tabelle 6.1 aufgeführten Variablen.

serverip	IP-Adresse des CATS-Servers
serverport	Port des CATS-Servers
sourceip	IP-Adresse des Senders (reserviert)
sourceport	Port-Adresse des Senders (reserviert)
command	CATS-Server-Kommando
destip	Ziel-IP-Adresse (reserviert)
destport	Ziel-Port-Adresse (reserviert)
id	Benutzername
password	Passwort des Benutzers
feedback	Rückmeldungsfeld
appletid	Applet-Identifizierung
reliability	Messwert für die Zuverlässigkeit
proficiency	Messwert für den Schwierigkeitsgrad
time	Zeitbedarf
ok	Status-Rückmeldung

Tabelle 6.1: Variablen der Result-Klasse

Das Result-Objekt erwartet die drei KPIs (s. Abschnitt 5.2.1), die Authentifizierungsdaten sowie die Adresse des CATS-Servers als Eingabe. Sobald die Übertragung der Ergebnisse stattfinden soll, wird das gesamte Result-Objekt serialisiert und per TCP/IP zu dem CATS-Server geschickt. Dieser wertet die Eingaben aus und liefert eine Feedback-Meldung, die in das Objekt eingetragen wird. Anschließend wird das serialisierte Objekt wiederum zum Client zurückübertragen, und die Feedback-Meldung wird entsprechend angezeigt.

Abbildung 6.6 stellt diesen Vorgang graphisch dar. Die verschiedenen Proficiency-Level werden nach den Vorgaben, die in Abschnitt 5.2.1 vorgestellt wurden, von dem Entwickler der Übungsaufgabe festgelegt. Im Abschnitt 7.1 werden einige CATS-Applets vorgestellt und hierbei die Festlegung der jeweiligen Proficiency-Levels dargestellt. An dieser Stelle soll ein einfaches Beispiel vorgestellt werden. Unter der Annahme, ein CATS-Applet soll die Grundlagen der Bruchrechnung zum Gegenstand haben.

$$\frac{30}{5} = ? \quad (6.1)$$

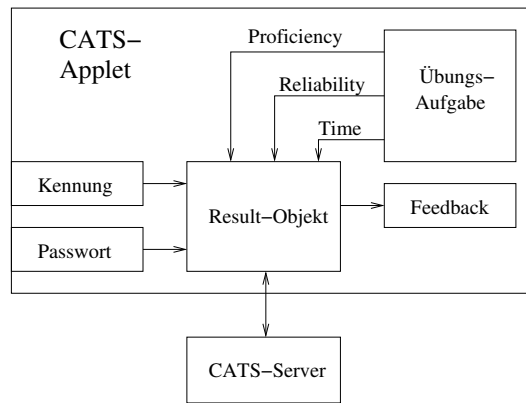


Abbildung 6.6: Die Integration des Result-Objektes in eine Übungsaufgabe

$$\frac{3 + 17}{2 \cdot 5} = ? \quad (6.2)$$

$$\frac{4 \cdot (3 + 6)}{2 \cdot (3 + 5)} = ? \quad (6.3)$$

$$\frac{4 \cdot (5 + 7)}{3 \cdot (12 + 2)} + \frac{6 \cdot (8 + 4)}{7 \cdot (9 + 5)} = ? \quad (6.4)$$

$$\frac{\frac{5 \cdot (12 + 45)}{8} \cdot (8 + 9)}{6 \cdot (7 + 4)} + \frac{6 \cdot (13 + 7)}{7 \cdot (12 + 9)} = ? \quad (6.5)$$

In diesem Beispiel hätte die Aufgabe 6.1 die, die einfache Division zweier Zahlen zum Gegenstand hat, den Schwierigkeitsgrad 1, die Aufgabe 6.2, die insbesondere einfache zusätzliche Rechenoperationen zur Bruchrechnung testet, hätte den Schwierigkeitsgrad 3, Aufgabe 6.3, die nun noch eine Multiplikation beinhaltet, den Schwierigkeitsgrad 5, Aufgabe 6.4, den Schwierigkeitsgrad 11 und die in diesem Beispiel komplizierteste Aufgabe 6.5 den Schwierigkeitsgrad 17. Die Zuordnung der Schwierigkeitsgrade hat sich in diesem Beispiel aus der Anzahl der zur Lösung erforderlichen Lösungsschritte ergeben.

Der Wert der Variablen "reliability" (Zuverlässigkeit) wird um eins für jede richtig gelöste Aufgabe erhöht und für jede falsch gelöste Aufgabe um ein erniedrigt.

Die Zeit wird in Millisekunden gemessen und als ganzzahliger Integerwert an das Result-Objekt übergeben.

Die Feedback-Meldung des Servers ist eine reine Textmeldung, die im Arbeitsbereich des CATS-Applets angezeigt werden soll.

Sollte später die Übermittlung der Ergebnisse auf andere Art und Weise erforder-

lich werden (z. B. bei der Verwendung einer Lernplattform), so muss lediglich diese Klasse verändert werden.

Prototypisch wurde als Gruppenaufgabe eine simulierte Gerichtsverhandlung (so genannter Moot-Court) realisiert. Hierbei existieren die Rollen (für die Studierenden) "Staatsanwalt", "Verteidigung", "Richter". Mit der Kennung der Studierenden wird vom CATS-Server jeweils einer dieser Rollen den Studierenden zugewiesen. Jeder dieser Rollen erhält einen Arbeitsbereich, der jeweils gleich aufgebaut ist. Dieser ist in Abbildung 6.7 dargestellt. Der Bereich "Case" wird durch den Initialisierungs-

The image shows a graphical user interface for a Moot-Court simulation. It features a vertical list of text input fields on the left, each with a label and a corresponding text area on the right. The labels are: ID, Password, Object ID, Case, Brainstorm, Structure, Opinion, Pleading, Review, and Messages. The text areas contain the following content: ID (empty), Password (empty), Object ID (empty), Case ("A schießt B tot, wie ist die Rechtslage?"), Brainstorm ("C: Der Fall scheint leicht zu sein. Morgen 19:00 MCU?"), Structure ("A -> 211 StGB - fehl. Mordm. A-> 212 StGB+"), Opinion ("A könnte sich des Totschlags gem. §211 StGB strafbar gemacht haben. Voraussetzung"), Pleading ("A wird wegen Totschlags zum Nachteil des B angeklagt. Die Tat ist strafbar gem. §211"), Review ("Der Verhandlung fehlt bisher der 'Pfiff'"), and Messages (empty). Below these fields are four buttons: InitApplet, Send_Object, Get_Object, and Transmit Results. At the bottom, there is a label "System Message:" followed by a large empty text area.

ID:	
Password:	
Object ID:	I
Case:	A schießt B tot, wie ist die Rechtslage?
Brainstorm:	C: Der Fall scheint leicht zu sein. Morgen 19:00 MCU?
Structure:	A -> 211 StGB - fehl. Mordm. A-> 212 StGB+
Opinion:	A könnte sich des Totschlags gem. §211 StGB strafbar gemacht haben. Voraussetzung
Pleading:	A wird wegen Totschlags zum Nachteil des B angeklagt. Die Tat ist strafbar gem. §211
Review:	Der Verhandlung fehlt bisher der "Pfiff"
Messages:	

InitApplet Send_Object Get_Object Transmit Results

System Message:

Abbildung 6.7: Der Arbeitsbereich der Rollen im Moot-Court

prozess fest vorgegeben. Alle Teilnehmer können hierauf lesend zugreifen. Der Bereich "Pleading" aller Rollen ist jeweils für alle Teilnehmer lesbar, allerdings können nur die jeweiligen Rollen diesen Bereich beschreiben. Die Bereiche "Brainstorm", "Structure", "Opinion" sind jedoch vertraulich und nur den jeweiligen Rollen schreibend, wie lesend zugänglich. Die Realisierung der entsprechenden Rollen wurde bereits in Abschnitt 5.10 vorgestellt.

6.3 Das CATS-Autorenwerkzeug für projektbezogene Aufgaben

Zusätzlich zu dem Multiple-Choice-Editor (s. Abschnitt 5.5.1), wurde entsprechend der Architektur des CATS-Aufgabeneditors für projektbezogene Aufgaben (s. Abschnitt 5.5.2) eine Java-Anwendung programmiert, die es auch ohne programmier-technische Kenntnisse ermöglicht, einfache, dynamische projektorientierte Aufgaben zu generieren.

Hierbei kam die modifizierte Netzplantechnik zum Einsatz. Der Lehrer kann zunächst auf einer Arbeitsfläche einen "Soll-Ablauf" durch Blasen, die bestimmte Vorgänge repräsentieren, und Verbindungen zwischen diesen darstellen. Die einzelnen Vorgänge können auf bestimmte Methoden und Variablen anderer Vorgänge zugreifen. Um dynamische Aufgaben programmieren zu können, ist es notwendig, dass bestimmte Werte durch Zufallszahlen vorbelegt werden können. Dies geschieht entsprechend durch eine vorprogrammierte Funktion. Es muss hierbei nur der gültige Zahlenbereich vorgegeben werden. Diese Art der Formatierung wird auch bei sehr vielen Tabellenkalkulationsprogrammen (z. B. Microsoft Excel) verwendet, so dass dieses Konzept der Zielgruppe vertraut sein sollte. Die Abbildung 6.8 stellt den Ablauf schematisch dar.

Die fertigen Aufgaben werden dann in einer entsprechend strukturierten XML-Datei abgespeichert. Sobald ein Lerner die entsprechende Übungsaufgabe auswählt, wird ein generisches Applet geladen, welches die entsprechende XML-Datei einliest und dann die Übungsaufgabe präsentiert.

Der Schwierigkeitsgrad eines Aufgabentyps wird vom Ersteller der Aufgabe fest vorgegeben. Die Zuverlässigkeit und die Zeit werden wie bereits oben vorgestellt ermittelt und dem Result-Objekt übergeben.

Nach Bearbeitung der Aufgabe werden die KPIs dem CATS-System übermittelt und eine entsprechende Feedback-Meldung vom Server zurückgemeldet und dem Lerner angezeigt.

6.4 Die Kommunikationsunterstützung

Die Kommunikationsunterstützung bei CATS erfolgt in verschiedenen Konstellationen und durch unterschiedliche Methoden. Zum einen ist die Peer-to-Peer-Kommu-

6 Die Implementierung des Communication and Tutoring Systems (CATS)

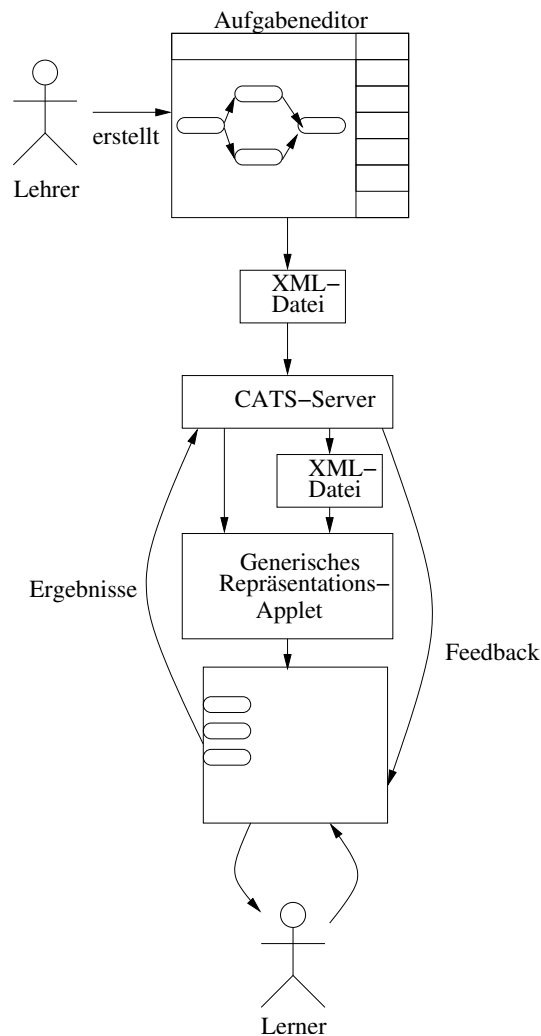


Abbildung 6.8: Die Erstellung einer dynamischen projektbezogenen CATS-Aufgabe

nikation von der Gruppenkommunikation zu unterscheiden. Für diese beiden Arten der Kommunikation werden entsprechend unterschiedliche Techniken eingesetzt.

6.4.1 Die Peer-to-Peer-Kommunikation

Die Peer-to-Peer-Kommunikation wurde sowohl durch eine H.323-kompatible Technik wie auch durch das Session Initiation Protocol(SIP) realisiert.

Die H.323 konforme Kommunikationsunterstützung

Eine sehr gute Integration in die Benutzeroberfläche ermöglicht die H.323-konforme Anbindung. Hierzu werden zunächst über ein PHP-Skript die zur Zeit verfügbaren Studierenden, die Kenntnisse bezüglich der ausgewählten Übungsaufgabe haben, angezeigt. Abbildung 6.9 zeigt diese Leiste. Hierüber können die Studierenden einen entsprechenden Kommunikationspartner anwählen. Der Verbindungsaufbau wird dann über die H.323-Anwendung (z. B. Netmeeting oder GnomeMeeting) realisiert.

The screenshot displays a web interface for the CATS-Awarenessfunction. It features a blue header area with the text "List of available students:". Below this, a table lists two students: 'test' with ID 1919 and IP 134.155.48.38, and 'liebig' with ID 81 and IP 134.155.48.38. At the bottom, there is a login section with input fields for ID (containing 'liebig'), Password (masked with asterisks), and Own IP (containing '134.155.48.38'). A message "Logged on" is displayed, and there are "Logon" and "Logoff" buttons.

ID:	Points	Call	IP-Address
test	1919		134.155.48.38
liebig	81		134.155.48.38

ID:	liebig
Password:	*****
Own IP:	134.155.48.38

Message: Logged on

Logon Logoff

Abbildung 6.9: Die CATS-Awarenessfunktion zur Peer-Kommunikation

6.4.2 Unterstützung der Gruppenkommunikation durch eine MCU

Um den Studierenden die Möglichkeit zu geben zeitgleich in einer Gruppe (> 2 Personen) in Form einer Videokonferenz zu kommunizieren, wurde eine so genannte Multipoint Control Unit (MCU) auf dem CATS-Server eingerichtet. Das verwendete System ermöglicht, dass sich bis zu vier Teilnehmer gleichzeitig in einem "Raum" in einer Audio/Videokonferenz¹ befinden. Die Anzahl der virtuellen Konferenzräume ist bei dem verwendeten System grundsätzlich nicht begrenzt.

Im Rahmen dieser Arbeit wurde die openMCU des Projekts "Open H.323" [Ltd03] verwendet. Die Teilnehmer können nun bestimmte "Räume" innerhalb des Systems belegen. Die Namen der Gruppenräume ist bei diesem System grundsätzlich frei wählbar. Im Rahmen des CATS-Systems gilt jedoch die Konvention, dass die Videokonferenz-Räume die Namen der jeweiligen Gruppen tragen. Dies ermöglicht die direkte Zuordnung und erleichtert auch die direkte Anwahl.

6.4.3 SIP-Integration

Eine weitere Möglichkeit der Kommunikation wurde durch das Session Initiation Protocol (SIP) geschaffen. Zu diesem Zweck wurde ein so genannter Registrar-Server auf dem CATS-Server eingerichtet. Hierzu wurden die SIP-Tools des National Institute of Standards and Technology (NIST)[Nat04] verwendet. Die Integration wurde im Rahmen einer Studienarbeit [Bei03] realisiert und wird im Folgenden beschrieben.

Das entwickelte Java-Programm bietet die Möglichkeit, einen Anruf zu einem Benutzer auf der Basis des Benutzernamens oder aufgrund seines Leistungsstandes herzustellen. Die Kontaktaufnahme geschieht in zwei Phasen: In der ersten Phase wird das Ziel lokalisiert, in der zweiten Phase geschieht der eigentliche Verbindungsaufbau. Im ersten Schritt wird der CATS-Server nach derjenigen Person befragt, die im Moment verfügbar ist und für eine gegebene Aufgabe bisher die beste Leistung erbracht hat. Der CATS-Server liefert nun die entsprechende SIP-Adresse zurück. Diese wird an die Client-Anwendung übermittelt. Der Client leitet sie daraufhin an den User Agent (im Rahmen dieses Projektes wurde der MSN-Messenger gewählt) weiter. Dieser verfügt über eine API, die über C/C++ und VB-Skript angesprochen werden

¹Die Anzahl der Audio-Teilnehmer ist unbegrenzt

kann. Der User Agent führt nun einen entsprechenden SIP-Anruf durch. Hierzu baut er zunächst zu dem SIP-Proxy des CATS-Systems eine Verbindung auf, er erhält von dort die aktuell gültige IP-Adresse des Empfängers und wickelt dann gemäß dem SIP-Protokoll (siehe hierzu Abschnitt 5.3.2) den Anruf ab.

Zu Beginn der Arbeit war die Verbreitung von SIP-konformen Produkten noch sehr selten. In der Zwischenzeit hat sich SIP im Bereich der VoIP-Technik etabliert und es gibt von fast allen bekannten Herstellern eine große Produktvielfalt.

6.5 Die CATS-Administration

Die CATS-Administration gliedert sich in zwei Bereiche: Die Systemadministration und den Benutzerbereich. Beide Systeme wurden mit Hilfe von PHP-Programmen realisiert und können über einen beliebigen Web-Browser gesteuert werden.

6.5.1 Die Systemadministration

Die wesentlichen Aufgaben der Systemsadministration werden durch das so genannte Exercise Administration and Controlling System (EACS) erfüllt: Die Appletverwaltung, die Pflege der Benutzerdaten und das Controlling der integrierten Übungsaufgaben. Die gesamte Verwaltung erfolgt durch ein Web-Interface, welches mit Hilfe der Skriptsprache PHP entwickelt wurde.

Die Appletverwaltung

Die Appletverwaltung registriert zunächst neu entwickelte Applets im System. Hierbei werden verschiedene Metadaten wie z. B. der Name, der/des Autorin / Autoren, curriculare Einordnung, Vorgabewerte bezüglich der Leistungsdaten eingegeben. Zudem werden noch ergänzende Materialien (z. B. umfangreichere Dokumentation, Ausarbeitungen) zugeordnet.

Nach der erfolgten Registrierung stehen die Applets in CATS zur Verfügung. Eine weitere Besonderheit ist die Öffnung der Quelltexte für wissenschaftliche und nicht kommerzielle Zwecke. Ein Nebeneffekt ist hierbei, dass die Studierenden dadurch auch die Wirkung und Verantwortung erfahren, die mit einer wissenschaftlichen Veröffentlichungen verbunden ist. Gerade dies ist ja ein Lernziel der universitären Ausbildung, welches oft nicht genügend gelehrt wird. Nachdem externe Anwender sich mit den entsprechenden Lizenzbedingungen einverstanden erklärt haben,

die unter anderem bei eigener Verwendung den zwingenden Hinweis auf die ursprüngliche Arbeit vorsehen, können sie auf die entsprechenden Quelltexte sowie die vertiefenden Materialien zugreifen.

Die Benutzerverwaltung

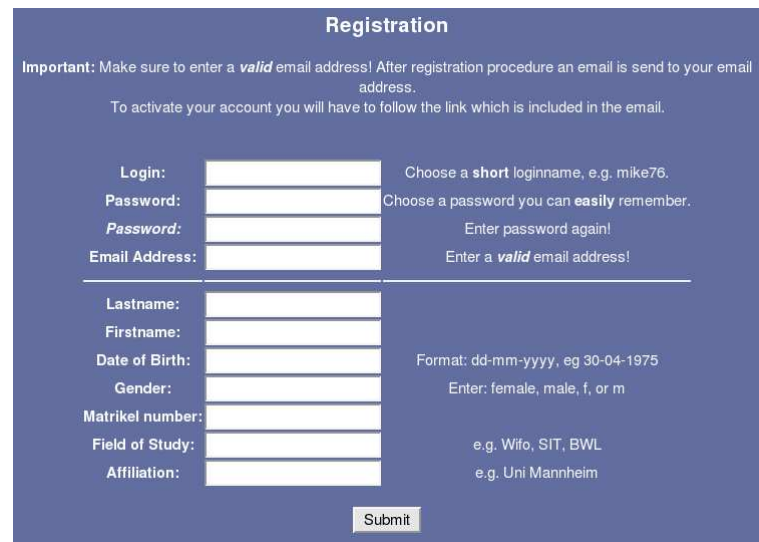
Neben der Benutzerverwaltung durch EACS ermöglicht eine ODBC-Anbindung in bekannten Office-Produkten (z. B. Excel, Access) die direkte Bearbeitung der Benutzerdaten.

Controlling-Funktion

Eine Funktionalität, die insbesondere für die Dozierenden gedacht ist, ist die Möglichkeit eines begleitenden Controllings. Sie können sich jederzeit ein umfassendes Bild über den Leistungsstand aller Studierenden machen. Hierbei können die Auswertungen auch bis auf die Gruppenebene aggregiert werden.

6.5.2 Der Benutzerbereich

Der Benutzer kann sich über das Web-Interface selbst registrieren und sich über seinen eigenen Leistungsstand informieren. Zu diesem Zweck wurde ein eigenes Registrierungsverfahren realisiert: Die Studierenden haben zunächst ihre Daten in ein Web-Formular einzugeben. Dieses ist in Abbildung 6.10 dargestellt. Diese werden dann durch die Anwendung in der Tabelle *Students* eingetragen, wobei vermerkt wird, dass die betreffenden Studierenden noch ihre Anmeldung bestätigen müssen. Danach verschickt das Programm eine E-Mail-Nachricht, die zum einen noch einmal den Benutzer über die Registration im CATS-System informiert und zum anderen eine Bestätigung anfordert. Zu diesem Zweck enthält die E-Mail-Nachricht einen verschlüsselten Freischaltcode. Sobald der Benutzer diesen Freischaltcode aktiviert, wird dies im entsprechenden Tabelleneintrag vermerkt, und der Benutzer kann das CATS-System verwenden. Dies ist notwendig, um ein Mißbrauch des Systems zu verhindern. Somit wird zumindest sichergestellt, dass der Benutzer eine gültige E-Mail-Adresse besitzt.



Registration

Important: Make sure to enter a *valid* email address! After registration procedure an email is send to your email address.
To activate your account you will have to follow the link which is included in the email.

Login:	<input type="text"/>	Choose a short loginname, e.g. mike76.
Password:	<input type="password"/>	Choose a password you can easily remember.
Password:	<input type="password"/>	Enter password again!
Email Address:	<input type="text"/>	Enter a valid email address!

Lastname:	<input type="text"/>	
Firstname:	<input type="text"/>	
Date of Birth:	<input type="text"/>	Format: dd-mm-yyyy, eg 30-04-1975
Gender:	<input type="text"/>	Enter: female, male, f, or m
Matrikel number:	<input type="text"/>	
Field of Study:	<input type="text"/>	e.g. Wifo, SIT, BWL
Affiliation:	<input type="text"/>	e.g. Uni Mannheim

Abbildung 6.10: Die CATS-Studierenden Registration

6.6 Anbindung von CATS an Lernplattformen

Da CATS auch mit einer Lernplattform verwendet werden soll, bestehen mehrere grundsätzliche Möglichkeiten einer Integration. Diese reichen von einfachen Links, zu denen in der Lernplattform lediglich die Meta-Daten verwaltet werden, bis hin zu einer direkten Anbindung, in der sowohl die Autorisierungsinformationen wie auch die Ergebnisse der Aufgaben in die Lernplattform übernommen werden können. Auch ist es, soweit die Lernplattform einen entsprechenden Export der Daten unterstützt die bestehenden Kennungen einzulesen und damit den Lernplattform-Nutzern die Verwendung des CATS-Systems zu ermöglichen.

6.6.1 Anwendung des CATS-Integrationsmodells

Wie bereits in Abschnitt 5.9.1 dargestellt, verwendet CATS ein dreischichtiges Modell. In jeder Schicht wäre eine Integration zu einer bestehenden oder geplanten Lernplattform möglich. Die Möglichkeiten reichen von der Integration auf der Ebene des DBMS über die Integration auf der Anwendungsebene bis hin zur Entwicklung eines eigenen Lernplattformmoduls, welches die existierenden Applets verwendet. Wie bereits dargelegt, ist CATS als offenes System konzipiert, somit ist eine Integration grundsätzlich auf all diesen Ebenen problemlos möglich. Allerdings müssen die Lernplattformen selbst für eine Integration ausgelegt sein und entsprechende

Schnittstellen zur Verfügung stellen. Dies ist, wie ebenfalls bereits dargelegt, bei sehr vielen Systemen der Fall. Da jedoch die Beschaffung, Installation und Konfiguration einer Lernplattform mit einem erheblichen Ressourcenbedarf verbunden sind, konzentriert sich die folgende Beschreibung einer exemplarischen Integration auf das .LRN-System der Universität Mannheim.

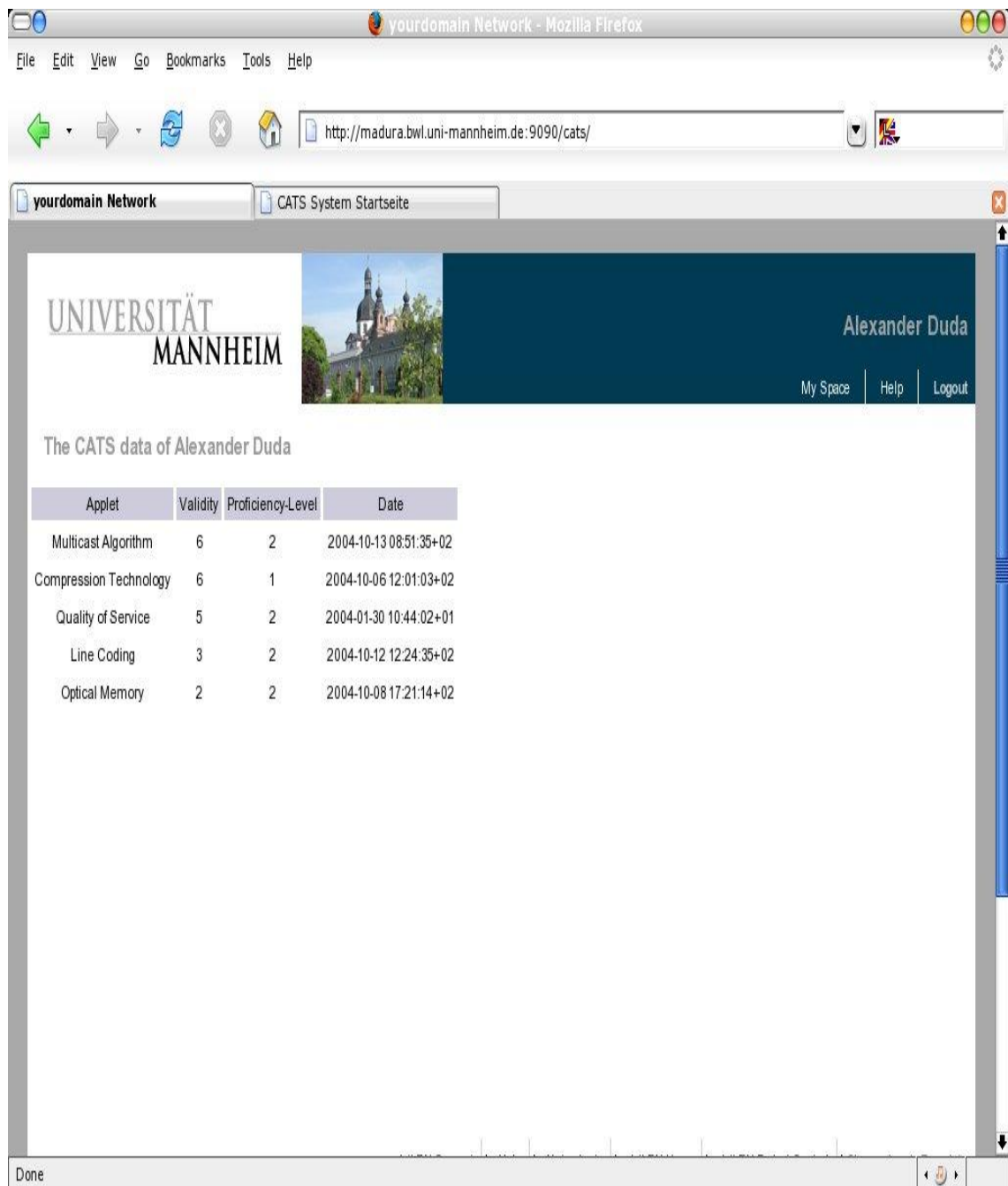
6.6.2 Anbindung von CATS an .LRN

Da .LRN selbst die gleiche relationale Datenbank verwendet wie CATS, kann bereits in der innersten Schicht von CATS, der Datenbankschicht, eine Integration erreicht werden. Dies wurde jedoch aus Gründen der gewünschten Systemunabhängigkeit nicht realisiert. Stattdessen wurde zunächst ein Teil der CATS-Tabellen identisch im .LRN-System angelegt. Auf diese können nun die einzelnen .LRN-Module zugreifen. Um die Ergebnisse des CATS-Systems in das .LRN-System zu übertragen, werden die entsprechenden Tabelleneinträge aus dem CATS-DBMS in das .LRN-DBMS dupliziert. Diese Duplikation erfolgt durch eine entwickelte Java-Applikation, die sowohl Zugang zu der CATS-Datenbank wie auch Zugang zu der .LRN-Datenbank hat. Zunächst werden alle vorhandenen CATS-Einträge in der .LRN Datenbank gelöscht und dann werden komplett alle aktuellen CATS-Einträge dupliziert. Hierdurch ist die Konsistenz der beiden Systeme zunächst sichergestellt.

Damit die CATS-Ergebnisse auch in der .LRN-Umgebung verwendet werden, wurde im Rahmen der Integration eine .LRN-Seite entwickelt, welche es den Studierenden erlaubt, ihre aktuellen CATS-Ergebnisse einzusehen. Hierbei wird der Anmelde-name an das entwickelte Modul übermittelt und eine entsprechende SQL-Abfrage ausgeführt. Die Abbildung 6.11 stellt die entsprechende Seite dar.

6.7 Zusammenfassung

Das CATS-System wurde implementiert hierbei wurde darauf geachtet, weitgehend Standardsoftware zu verwenden, um eine leichtere Übertragbarkeit des Systems (insbesondere der Server-Komponenten) sicherzustellen. Auch eine Anbindung an bestehende Lernplattformen kann ohne großen Aufwand vorgenommen werden. Das System wurde im produktiven Lehrbetrieb, in verschiedenen Lehr-/Lernszenarien eingesetzt und hierbei evaluiert.



yourdomain Network - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://madura.bwl.uni-mannheim.de:9090/cats/

yourdomain Network CATS System Startseite

UNIVERSITÄT MANNHEIM

Alexander Duda

My Space Help Logout

The CATS data of Alexander Duda

Applet	Validity	Proficiency-Level	Date
Multicast Algorithm	6	2	2004-10-13 08:51:35+02
Compression Technology	6	1	2004-10-06 12:01:03+02
Quality of Service	5	2	2004-01-30 10:44:02+01
Line Coding	3	2	2004-10-12 12:24:35+02
Optical Memory	2	2	2004-10-08 17:21:14+02

Done

Abbildung 6.11: Die CATS/.LRN-Integration: Ergebnis-Anzeige

7 Die Evaluierung des CATS-Systems

CATS wurde in verschiedenen Fächern und in verschiedenen Lehr-/Lernszenarien empirisch erprobt.

Das folgende Kapitel stellt zunächst einige Beispiele von CATS-Applets aus unterschiedlichen Fächern mit unterschiedlichen Themenbereichen dar. Wie bereits in den einleitenden Kapiteln dargelegt, wurde CATS als offenes System konzipiert, so dass viele Möglichkeiten existieren, neue Aufgaben zu erstellen und als CATS-Applets den Studierenden zur Verfügung zu stellen.

Die erste und im Sinne des Konstruktivismus speziell für die Informatik-Ausbildung sinnvolle Möglichkeit besteht darin, Studierende selbst CATS-Aufgaben in der Programmiersprache Java erstellen zu lassen. Hierbei werden sie durch ein Startpaket, bestehend aus der Java-CATS-Klasse "Results", einer Beispielaufgabe und einem einfachen Testserver, welcher die Messwerte zurückgibt, unterstützt. Dadurch können sie im Sinne des "Lernens durch Nachahmung" (s. Abschnitt 2.4.2) zunächst selbst kleine Übungsaufgaben erstellen und diese dann entsprechend den jeweiligen Anforderungen erweitern.

Diese Möglichkeit wurde im Rahmen von Studien- und Bachelorarbeiten genutzt, die im Folgenden dargestellt werden.

Andererseits können mit Hilfe der CATS-Autorenwerkzeuge können auch Nicht-Informatiker selbst Aufgaben erstellen und diese in das System integrieren. Hierbei gibt es bisher, wie in Abschnitt 5.5 dargestellt, zwei Möglichkeiten: Den Multiple-Choice-Editor und den Aufgabeneditor für projektorientierte Aufgaben.

Des Weiteren ist es möglich, bereits existierende Applets (z. B. Animationen) unter Verwendung der angebotenen Java-Klassen in das System zu integrieren.

7.1 Anwendungen aus dem Bereich der Informatik

Die Informatik ist ein mathematisch/technisch orientiertes Fach. Die verschiedenen Übungsaufgaben aus diesem Fachgebiet haben daher zum Ziel, den Studierenden verschiedenen technische Erkenntnisse, Zusammenhänge, Methoden und Algorithmen beizubringen. Hierbei ist insbesondere die interaktive Eingabe von Lösungen, die über das Multiple-Choice-Paradigma hinausgeht, von Bedeutung. Bei komplexen Themen wurde neben einem Übungsbereich zudem eine Simulationsumgebung realisiert, um den Studierenden eine freiere Auseinandersetzung mit dem Stoffgebiet zu ermöglichen.

7.1.1 CATS-Applets zum Thema Rechnernetze

Das Fach Rechnernetze ist direkt am Lehrstuhl für Praktische Informatik IV angesiedelt. Es handelt sich hierbei um eine Veranstaltung, welche aus einer Vorlesung im Umfang von 4 Semesterwochenstunden (SWS) und einer begleitenden Übung im Umfang von 2 SWS besteht. Im Anschluss an die Vorlesung, meist im folgenden Semester, wird ein optionales Praktikum angeboten.

Die Themen orientieren sich an den Schichten des ISO/OSI-Referenzmodells. Für CATS wurden insbesondere folgende Themen aufbereitet:

- Signalcodierung auf Leitungen
- der Token Ring
- Zugriffsverfahren im WLAN gemäß IEEE 802.11
- Routing mit Hilfe des Dijkstra-Algorithmus für Kürzeste Pfade
- Simulation und Übung von Multicast-Algorithmen
- die Überlast-Kontroll-(Congestion-Control)-Mechanismen in TCP.

Das CATS-Applet zur Signalcodierung von Bitströmen auf Leitungen

Ein Applet zum Erlernen der Signalcodierung von Bitströmen auf Leitungen wurde von Sasan Safai im Rahmen einer Studienarbeit entwickelt [Saf03]. Die Lerner haben die Aufgabe, Bit-Ströme in physikalische Signal-Pegel umzusetzen bzw. gegebene Signalpegel entsprechend den Vorgaben umzukodieren. Hierbei gibt das Ap-

plet zunächst eine zufällige Bitfolge vor und verlangt in der leichten Schwierigkeitsstufe, eine einfache Leitungskodierung (z. B. die so genannte Non-Return-To-Zero-Codierung) durchzuführen. Der Schwierigkeitsgrad wurde, als einfachste Aufgabe mit dem Wert 1 belegt. Mit dem Anstieg des Schwierigkeitsgrades werden nun die Kodierungsvorschriften immer komplexer. Die schwierigste Codierung, die abgeprüft wird, ist die differentielle Manchester-Codierung. Diese hat dann den Schwierigkeitsgrad 4.

Die Abbildung 7.1 stellt das Applet zur Leitungskodierung dar. Das Besondere an

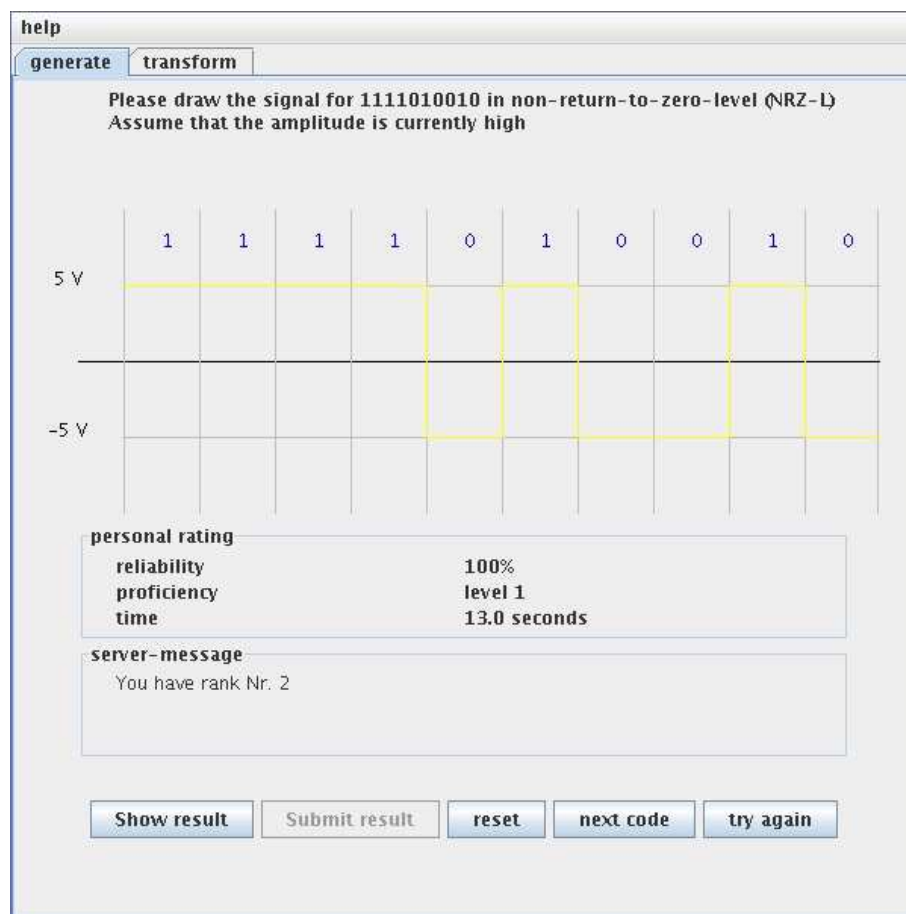


Abbildung 7.1: Das Line-Coding Applet

diesem Applet ist, dass die richtige Lösung gezeichnet werden muss. Mit Hilfe einer speziellen "Fang-Funktion", die die Zeichenfläche entsprechend rastert, werden die Lerner hierbei durch das Applet unterstützt. Die Anzeige einer Stoppuhr wirkt zusätzlich als Motivator und trainiert die Studierenden auch unter Zeitdruck konzentriert zu arbeiten.

Übungsapplet zum Thema Token Ring

Ein Thema, welches immer noch für Rechnernetze aktuell ist, wenngleich die praktische Relevanz in den letzten Jahren deutlich nachgelassen hat, ist der Token Ring. Da jedoch das Grundprinzip, des Medienzugangs im LAN ohne Kollisionen, von grundsätzlichem Interesse ist, wird das Thema nach wie vor in der Rechnernetze-Vorlesung behandelt.

Das Applet von Lidia Konurov (s. [Kon04]) behandelt das Thema Token Ring. Den Aufgaben liegt das Normal-Token-Release-Verfahren zugrunde. Außerdem werden weitere Vereinfachungen gemacht: Es kann nun ein Paket von einer Station gesendet werden, einzelne Pakete entsprechen einer 10ms-Paketgröße. Die Stationen können nur die ersten vier Prioritätsstufen nutzen, alle Stationen haben verschiedene MAC-Adressen. Das Empfangen und Kopieren der Daten durch die Zielstation schlägt nicht fehl. Es werden von dem Applet die Themengebiete Grundprinzipien des Token Passing Steuerungsverfahrens, der Monitoring Mechanismus und der Prioritätsmechanismus behandelt.

Das Applet fordert im einfachsten Fall die Studierenden dazu auf, nach einer dynamisch generierten Aufgabenstellung grundlegende Verfahren im Token mit Hilfe von Auswahlboxen aufzuzeigen. Dies ist in Abbildung 7.2 dargestellt. Mögliche Verfahren wären u. a. : Station X kopiert und wiederholt ein Frame, Station Y generiert ein neues Token, usw. Auf der linken Seite befinden sich die genauen Daten des Ringes und die Angaben über die verschiedenen Anforderungen der einzelnen Stationen, im oberen Bereich die genaue Aufgabenstellung und in der Mitte ein Abbild des konkreten Ringes.

In den schwierigeren Stufen müssen dann die Studierenden Token-Ring-spezifische Berechnungen durchführen. Diese erwarten allgemeinen Berechnungen, wie z. B. wie die maximale Rahmengröße in Abhängigkeit von Token Holding Time und Übertragungsgeschwindigkeit, oder nach der Speicherkapazität des Netzwerkes.

Zugriffsverfahren im Wireless Local Area Network gemäß IEEE 802.11

Die Zugriffsverfahren im Wireless Lan (WLAN), sind aufgrund der physikalischen Beschaffenheit des Mediums von denen der leitungsgebundenen Medien zu unterscheiden. Aufgrund der unterschiedlichen Sende- und Empfangsradien können insbesondere zwei Probleme mit WLAN entstehen, die in dieser Form bei leitungsgebundenen nicht auftreten: Das so genannte *Hidden Terminal Problem* und das *Exposed*

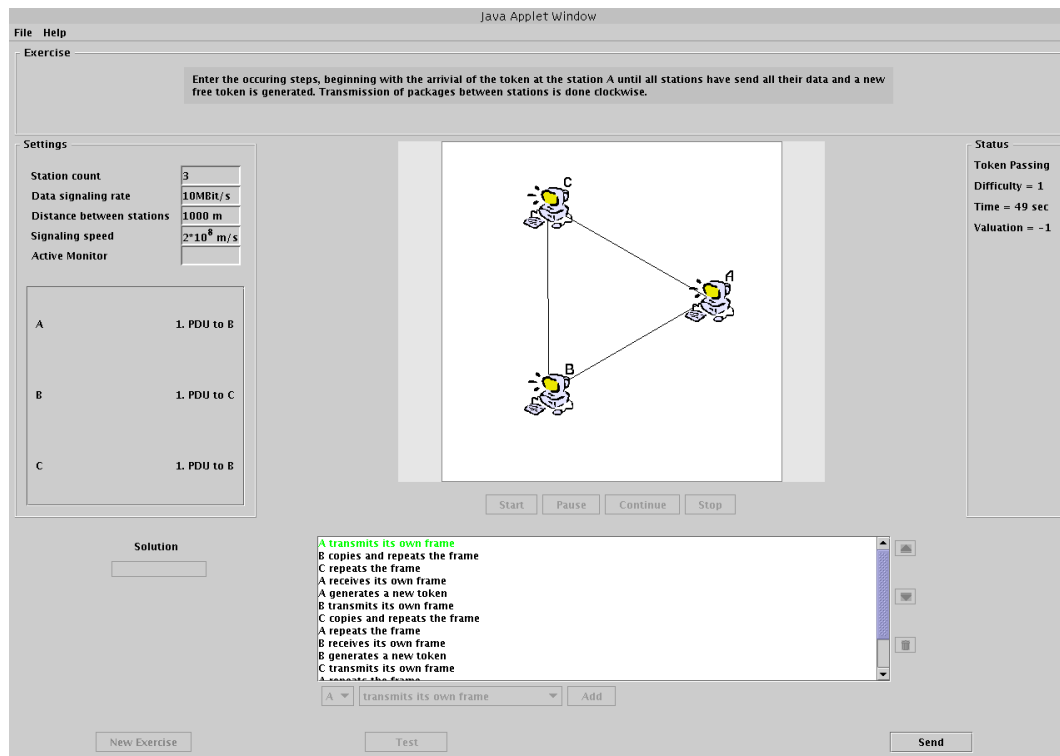


Abbildung 7.2: Das CATS-Applet zum Thema "Token Ring"

Terminal Problem.

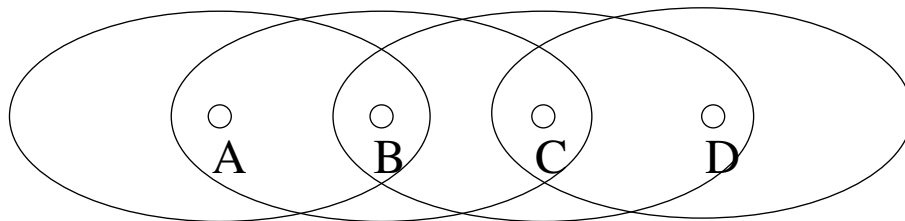


Abbildung 7.3: Probleme im Medienzugriff bei WLAN

Abbildung 7.3 stellt die Problematik bei WLAN-Netzen dar. Die Station A und B bzw. B und C können sich zwar gegenseitig hören, aber die Stationen A und C können ihre Signale gegenseitig nicht empfangen. Dies führt einerseits zu dem Problem, dass wenn die Station A zur Station B sendet, während gleichzeitig die Station C zu B sendet eine Kollision bei B stattfindet und die sendenden Stationen A und C hierüber zunächst keine Informationen erhalten (so genanntes *Hidden Terminal Problem*). Andererseits würde in dem Fall das B an A sendet, die Station C die sich ja

7 Die Evaluierung des CATS-Systems

ebenfalls in der Reichweite von B befindet eine etwaige Sendung an D zurückstellen, obwohl dies ohne eine Kollision möglich wäre. Dieses Problem bezeichnet man als *Exposed Terminal Problem*.

Um neben der allgemeinen Kollisionsvermeidung auch diese beiden Probleme zu lösen existieren im WLAN-Protokoll verschiedene Verfahren. Das Distributed Foundation Wireless Medium Access Control (DFWMAC) verwendet hierzu drei Verfahren: Das Distributed Coordination Function -Carrier Sensing Multiple Access with Collision Avoidance (DFWMAC-DCF CSMA/CA), das DFWMAC-DCF mit dem Request-to-Send / Clear-to-Send (RTS/CTS)-Mechanismus und das DFWMAC-Point Coordination Function (PCF)-Verfahren. Diese drei Verfahren sind Gegenstand des CATS-Applets von Michael Grauer [Gra05].

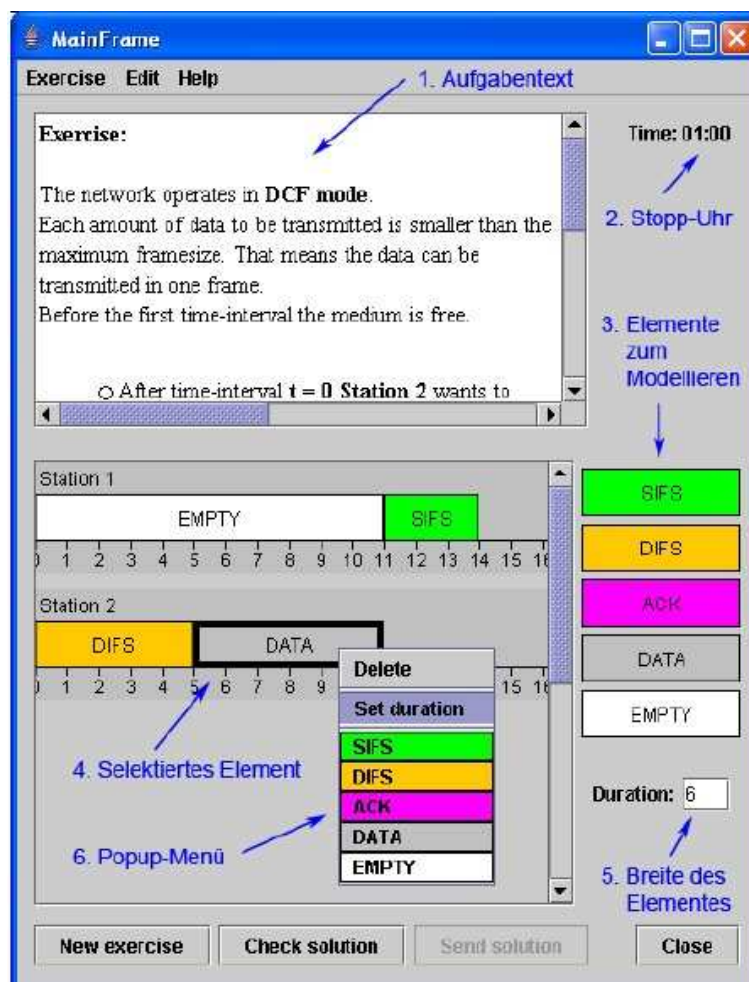


Abbildung 7.4: Das WLAN-Übungsapplet

Abbildung 7.4 stellt das Applet dar. Die Studierenden erhalten zunächst im oberen Bereich eine Aufgabenstellung präsentiert. Hierbei werden sowohl der verwendete Modus, wie auch die Sendewünsche der einzelnen Stationen aufgeführt. Das Protokoll vermeidet die o. g. Probleme mit prioritätsgesteuerten Senderahmen. Es werden drei so genannte Inter-Frame Spacings unterschieden:

Short Inter-Frame Spacing (SIFS) Diese Zeit warten alle Stationen in jedem Fall. Sendungen direkt an Anschluss dieses Rahmens haben die höchste Priorität. Clear to Send und Acknowledgements würden z. B. in diese Kategorie fallen.

Point Coordination Function Inter-Frame Spacing (PIFS) Diese Framezeit muss ein Sendewunsch mittlerer Priorität mindestens warten. Diese Zeit wird beim so genannten Polling bei Verwendung eines Access Points (AP) gewartet.

Distributed Coordination Function Inter-Frame Spacing (DIFS) Diese Zeit müssen alle Stationen mindestens warten, wenn sie in Konkurrenz zu den anderen Stationen treten will.

Die Studierenden sollen nun neben diesen Inter-Frame Zeiten noch Daten-Pakete und Wartezeiten zu den entsprechenden Zeiten in ein vorgegebenes Diagramm einzeichnen.

Es werden folgende Fälle abgefragt:

- Kommunikation in der DCF-Phase
- Kommunikation in der DCF-Phase unter Verwendung von Fragment Burst
- Kommunikation in der DCF-Phase mit RTS/CTS Signalen
- Kommunikation in der PCF-Phase

Der Schwierigkeitsgrad ergibt sich durch das gewählte Verfahren, der Anzahl der Stationen und die Anzahl der Sendewünsche.

Ein CATS-Applet zu dem Routing-Verfahren nach Dijkstra's Algorithmus für Kürzeste Pfade

Das Applet wurde von Markus Egenberger [Ege03] entwickelt. Durch dieses Applet sollen die Studierenden eines der grundlegenden Routing-Verfahren, den "Kürzesten Pfad" nach Dijkstra, erlernen. Das Applet bietet zunächst einen Tutoring-Mode

an, um den richtigen Umgang mit dem Aufgabentrainer zu üben. Hierbei wird eine fest vorgegebene Netzwerktopologie präsentiert, und anhand dieser werden die einzelnen Lösungsschritte erklärt.

Im Übungsmodus wird dann jeweils eine zufällige Netzwerktopologie vorgegeben, die Studierenden sollen nun den kürzesten Pfad einzeichnen. Hierbei sollen sie jedoch schrittweise, wie es auch dem Algorithmus entspricht, vorgehen. Die Abbildung 7.5 stellt den Aufbau des Applets dar. Die Studierende sehen auf der linken

Shortest Path Applet

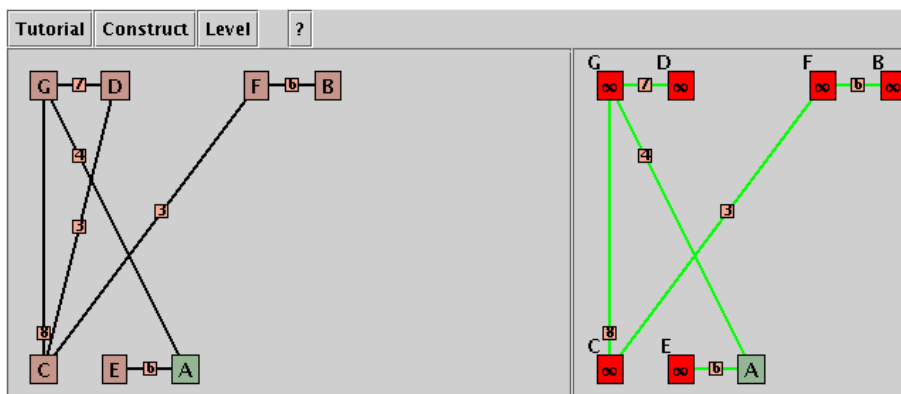


Abbildung 7.5: Das CATS-Applet zum Routing-Verfahren nach Dijkstra's Algorithmus für Kürzeste Pfade

Seite die gesamte Netzwerktopologie mit gewichteten Kanten, sie sollen dann auf der rechten Seite die einzelnen Verbindungen nacheinander einzeichnen, falls sie in einen Fehler entdecken, können die Studierenden auch wieder eine Verbindung entfernen.

Zum Schluss bleiben dann nur die kürzesten Routen, die zwischen allen Knoten bestehen, übrig. Der Schwierigkeitsgrad der Aufgabe bestimmt sich aus der Anzahl der Knoten und dem Grad der Vermaschung zwischen den Knoten.

Nach Abschluss der Aufgabe wird nicht nur die richtige Lösung angezeigt, sondern es werden auch die entsprechenden fehlerhaften Verbindungen markiert.

Ein CATS-Applet als Simulations- und Übungsapplet für Multicast-Routing-Verfahren

Insbesondere für die Übertragung von multimedialen Medienströmen (wie sie z. B. bei Videokonferenzen erzeugt werden) sind so genannte Multicast-Routing-Verfahr-

en von entscheidender Bedeutung, um die Netzwerklast zu reduzieren. Durch diese Verfahren werden Sender-Empfänger-Verbindungen nicht mehr Punkt-zu-Punkt aufgebaut, sondern über Multicast-Gruppenadressen Baum-strukturierte Routen erzeugt, die dann die gesamten Empfänger einer Multicast-Sitzung mit den übertragenen Informationen versorgen.

Aufgrund dieser Bedeutung ist das Themengebiet wichtiger Bestandteil der Rechnernetze- und Multimediatechnik-Vorlesung. In der Studienarbeit von Marc Hermann wurde zum einen eine Multicast-Simulationsumgebung entwickelt und zum anderen ein Multicast-Trainer realisiert [Her03a].

In der Simulationsumgebung können die Studierenden im Sinne des Konstruktivismus zunächst selbst Netzwerke konstruieren und austesten. Während der schrittweise ausführbaren Simulation sind die verschiedenen Informationspakete sichtbar dargestellt. Nach Abschluss der Simulation können dann statistische Informationen für jeden Knoten abgerufen werden. Abbildung 7.6 zeigt den zweiseitigen Aufbau

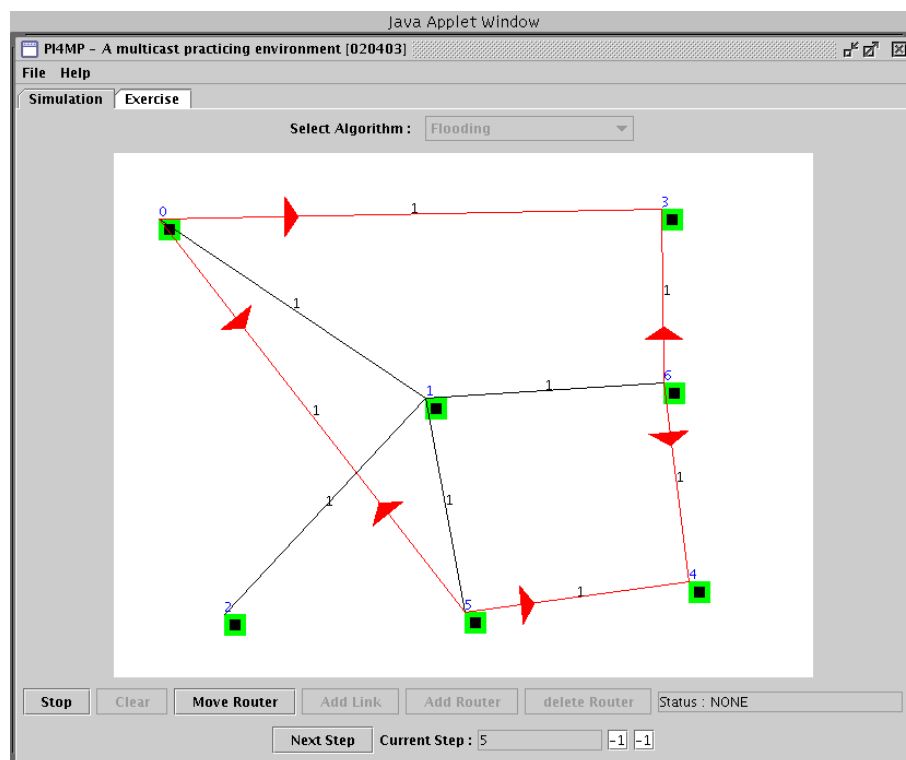


Abbildung 7.6: Das Multicast-Simulations- und Übungsapplet

des Applets. Die Studierenden müssen im Trainingsmodus Fragen zu dem Netzwerk beantworten. Im Bereich der Simulation werden die einzelnen Pakete sichtbar

gemacht. Folgende Multicast-Routing-Verfahren werden unterstützt:

- Flooding
- Truncated Reverse Path Broadcasting
- Protocol Independent Multicast Sparse Mode (PIM Sparse-Mode)

Das Trainingsapplet zur Überlast-Kontrolle in TCP

Die Überlast-Kontrolle (Congestion Control) in Netzwerken ist für einen effizienten und fairen Informationsaustausch von entscheidender Bedeutung. Aus diesem Grund werden entsprechende Verfahren in die Übertragungsprotokolle eingebaut. Die am weitesten verbreiteten Verfahren sind die, die im Rahmen des Transmission Control Protocol (TCP) verwendet werden. Erfahrungsgemäß bereitet dieses Themengebiet in Klausuren immer wieder große Schwierigkeiten. Aus diesem Grund wurde dazu ein CATS-Applet von Christine Kapp entwickelt[Kap02].

The screenshot shows a Java applet window titled 'Congestion-Control-Applet'. It has three tabs at the top: 'berechnen' (selected), 'auswerten', and 'zeichnen'. The 'berechnen' tab contains three sections:

- Einstellungen:** Two input fields. 'Anzahl der Aktionen:' has the value '3'. 'maximale Rundenzahl:' has the value '50'. To the right is a button labeled 'Aufgabe erzeugen'.
- Aufgabe:** A text area containing the following text:
Nach 7 Runden kommt es zu einem TDACK.
Nach 20 Runden kommt es zu einem TDACK.
Nach 45 Runden kommt es zu einem TDACK.
Wie groß ist das Congestion Window nach 50 Runden?
Wie groß ist ssthresh nach 50 Runden?
- Ergebnis:** Two input fields. 'CongestionWindo' has the value '10'. 'ssthresh' has the value '64'. To the right is an 'OK' button.

Abbildung 7.7: Das Congestion-Control-Applet: Berechnungsaufgabe

Bei der Realisierung des Applets wurde aus didaktischer Sicht verschiedenen Lernertypen Rechnung getragen: Ein Teil der Aufgaben kann berechnet werden (s. Abbildung 7.7), bei einem anderen Teil muss eine graphische Analyse vorgenommen

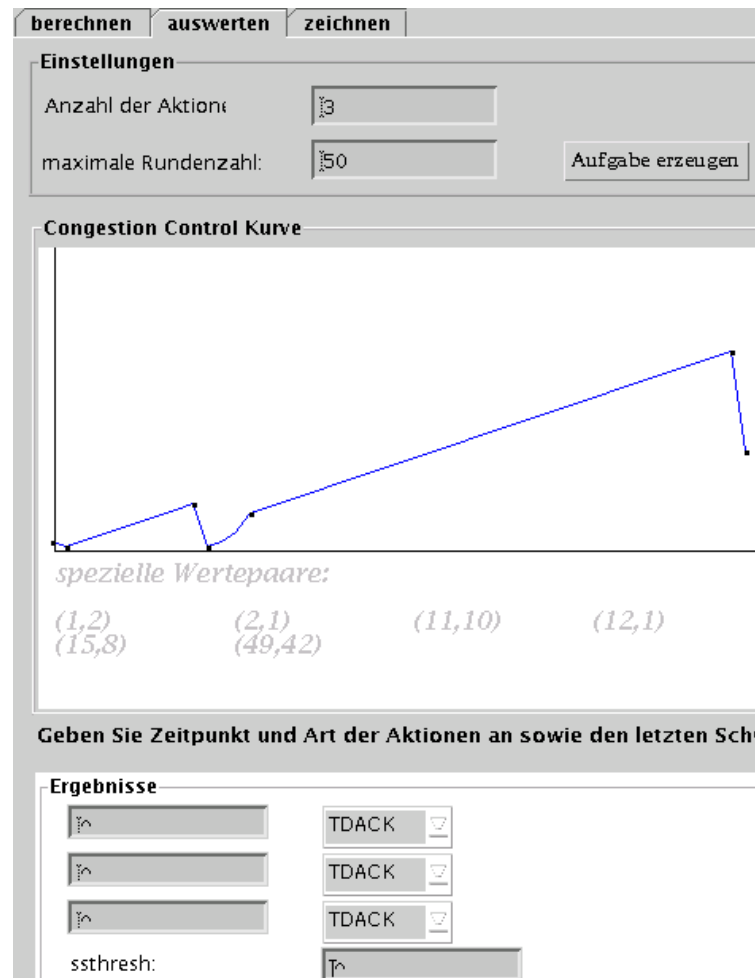


Abbildung 7.8: Das Congestion-Control-Applet: Analyseaufgabe

werden (s. Abbildung 7.8), und bei einer weiteren Kategorie erfolgt die Ergebniseingabe auf graphische Weise (s. Abbildung 7.9). Dies erfolgt dadurch, dass die Studierenden jeweils den Beginn und das Ende eines Abschnittes im Zeitverlauf, sowie die für diesen Abschnitt geltende Funktion (z. B. Slow-Start, lineares Wachstum, Timeout, etc.). für das Congestion Window eingeben müssen.

7.1.2 CATS-Applets zu Themen der Multimediatechnik

Ein weiterer Schwerpunkt der Lehrveranstaltungen des Lehrstuhls liegt im Bereich der Multimediatechnik. Hierzu finden regelmäßig eine Vorlesung sowie eine Übung im Umfang von 4+2 SWS statt. Diese Veranstaltungen werden ebenfalls durch ein optionales Praktikum ergänzt. Folgende Themengebiete konnten durch interaktive

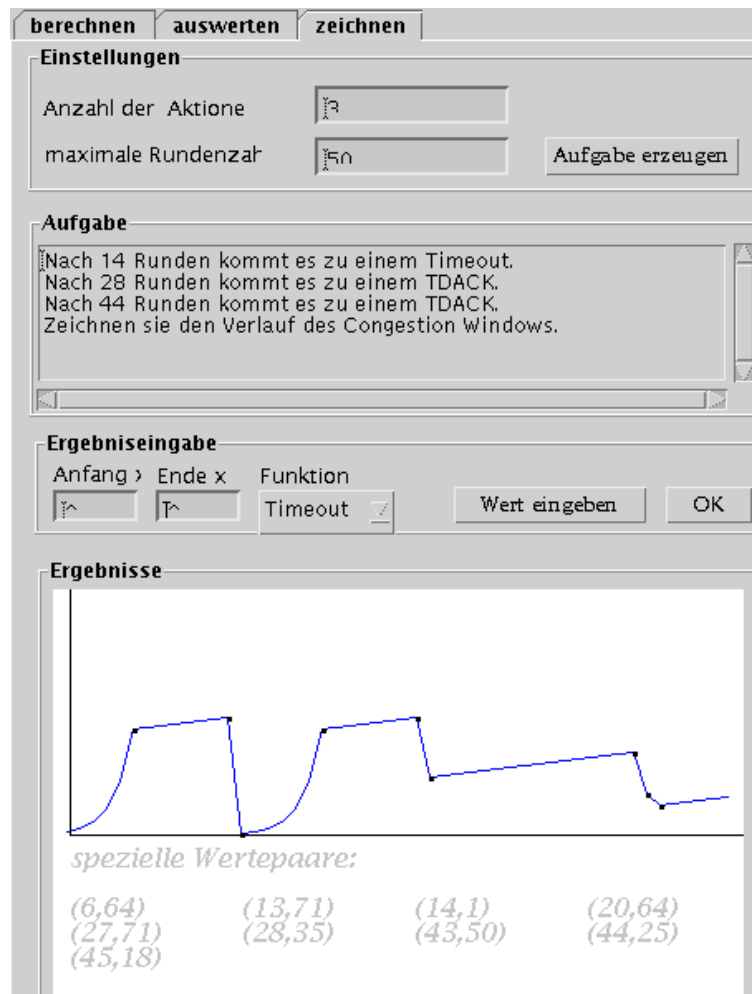


Abbildung 7.9: Das Congestion-Control-Applet: Zeichnung

CATS-Applets abgedeckt werden:

- Huffman-Codierung
- Fouriertransformation
- Medienserver
- Fehlerkorrekturverfahren bei optischen Medien
- Multicast-Verfahren
- Kosinus-Transformation.

Daneben wurde der restliche Stoff der Vorlesung Multimediatechnik mithilfe des Multiple-Choice-Applets (s. Abschnitt 5.5.1) durch fachbezogene Fragenkataloge abgedeckt.

Ein kognitives Tutoring-System zur Huffman-Kodierung

Die Huffman-Kodierung wurde von David A. Huffman 1952 erfunden [Huf52]. Es handelt sich hierbei um einen Algorithmus zur Erstellung von Bitcodes mit variabler Länge. Obwohl in der Zwischenzeit weitere Algorithmen zur Komprimierung erfunden wurden (z. B. Lempel-Ziv, arithmetrische Kodierung), ist die Huffman-Kodierung dennoch bis heute von Bedeutung, wie Bookstein in [BKR93] ausführt. Die Huffman ist z. B. gegenüber der arithmetrischen Kodierung schneller, sie ist intuitiver und bei Texten spart die arithmetrische Kodierung kaum mehr Platz ein. Für die Huffman-Codierung werden drei Algorithmen benötigt: Ein Algorithmus zur Erstellung des Huffman-Baums, einer zur Codierung des Datenstromes mit Hilfe des Huffman-Baums und ein Algorithmus zur Decodierung des komprimierten Datenstroms. Diese Algorithmen sind u. a. auch Gegenstand der Vorlesung Multimediatechnik.

Julia Peterwitz bearbeitet in ihrer Arbeit diese drei Algorithmen, die den Studierenden beigebracht werden sollen [Pet05]. Die Algorithmen werden im Folgenden beschrieben.

Der Algorithmus zur Erzeugung des Huffman-Baums lautet:

1. Bestimme die Häufigkeit der vorkommenden Zeichen;
2. Beschrifte die Zeichen mit ihrer Auftrittswahrscheinlichkeit und lege sie als Blattknoten an;
3. Suche zwei Knoten aus den kleinsten vaterlosen Knoten aus;
4. Erzeuge Vaterknoten mit der aufsummierten Wahrscheinlichkeiten der beiden Söhne;
5. Weise dem linken Pfad eine 0 und dem rechten eine 1 zu;
6. Gibt es noch mindestens 2 vaterlose Knoten, gehe zu Schritt 3, andernfalls ist der Algorithmus beendet.

7 Die Evaluierung des CATS-Systems

Der Algorithmus zur Codierung lautet:

1. Initialisiere die Daten-Warteschlange;
2. Nimm zu codierendes Zeichen aus der Warteschlange;
3. Gehe von der Wurzel bis zum Blattknoten des Zeichens entlang der Pfade und speichere die dabei passierteten Null- oder Eins-Bits;
4. Falls die Warteschlange nicht leer ist, gehe zu Schritt 2, andernfalls ist der Algorithmus beendet.

Der Algorithmus zur Decodierung lautet:

1. Initialisiere die Bitfolgen-Warteschlange;
2. Starte an der Wurzel des Huffman-Baums;
3. Wenn die Bitfolgen-Warteschlange nicht leer ist, lese Bit aus der Warteschlange, andernfalls ist der Algorithmus beendet;
4. Folge dem Pfad des gelesenen Bits (z. B. 0 nach links, 1 nach rechts)
5. Wenn der nächste Knoten ein Blattknoten ist, speichere das darin enthaltene Zeichen, andernfalls gehe zu 2., andernfalls gehe zu 3.

Die Studierenden haben genau nach diesen Algorithmen vorzugehen.

Abbildung 7.10 stellt das Applet dar. Im oberen Bereich wird ein zufällig ausgewähltes Wort aus einem Wörterbuch dargestellt. Die Studierenden haben nun Schritt für Schritt die entsprechenden Algorithmen durchzugehen. Im Fehlerfall erhalten sie als Besonderheit bei diesem CATS-Applet eine angepasste Rückmeldung und Hinweise zur richtigen Bearbeitung.

Die Fouriertransformation

Die Fouriertransformation bildet die Grundlage für die Signalverarbeitung und ist damit ein grundlegender Bestandteil der Vorlesung Multimediatechnik. Der Wechsel von Funktionen zu Distributionen wie auch die unterschiedliche Interpretation

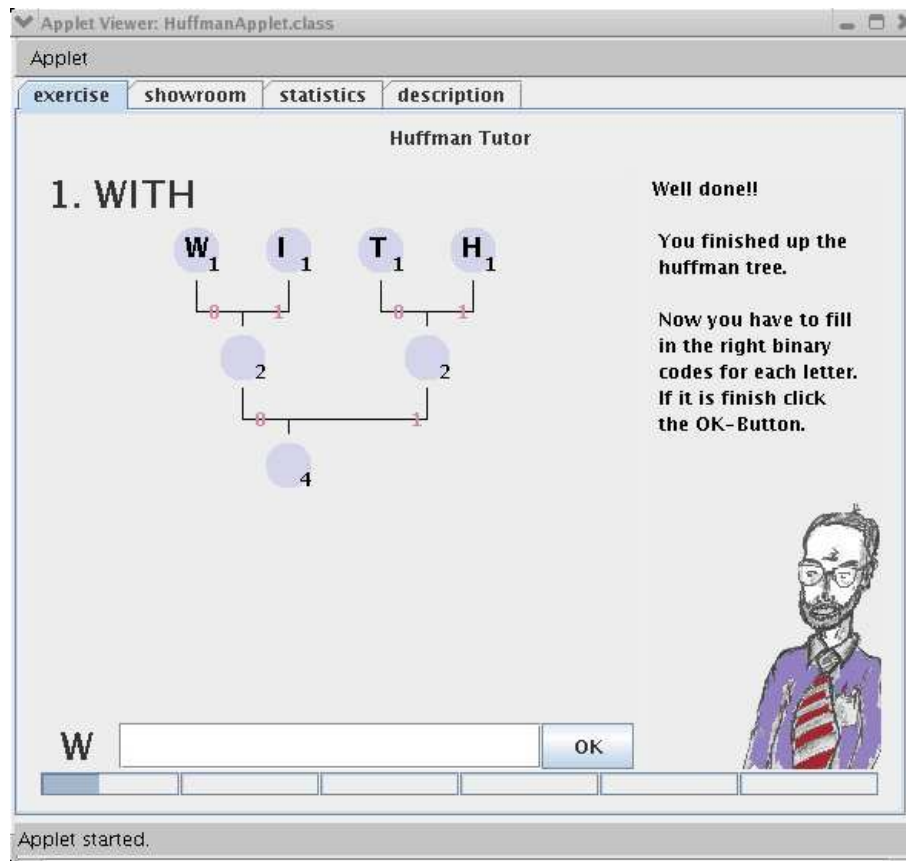


Abbildung 7.10: Der Kognitive Tutor zur Huffman-Codierung

von Signalen machen jedoch den Studierenden immer wieder sehr große Schwierigkeiten. Mit Hilfe des Applets von Christiane Adler [Adl02] können Grundlagen der Fouriertransformation erlernt werden. Insbesondere sind hierbei einige einfache Funktionen und ihre Transformation sowie die mit den Transformationen verbundenen Rechenvorschriften (Kommutativität, Faltung etc.) Lernziele dieses Applets. Die sofortige graphische Interpretation erlaubt den Studierenden ein vertieftes Verständnis der komplexen Materie. Abbildung 7.11 stellt die Übungsumgebung des Applets dar. Daneben besteht im Applet noch die Möglichkeit, einzelne Funktionen/Distributionen zu transformieren und sich das Ergebnis anzeigen zu lassen. Zu diesem Zweck werden die Eingaben der Studierenden geparkt und interpretiert. Die einfachste Schwierigkeitsstufe, sind gerade Funktionen, die nicht aus mehrere Funktionen zusammengesetzt sind, zum Beispiel eine einfache Kosinusfunktion oder ein Rechteckimpuls. Additionsfunktionen erhöhen den Schwierigkeitsgrad, eine weitere Erhöhung des Schwierigkeitsgrades findet durch Faltungen und Multi-

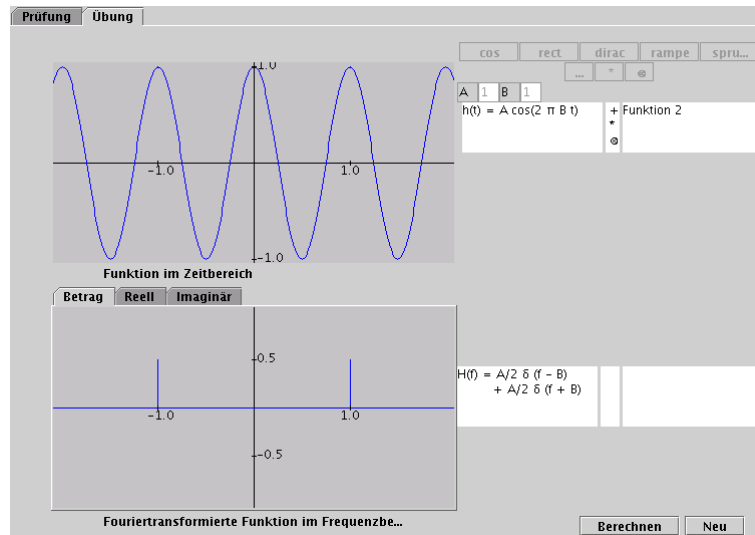


Abbildung 7.11: Die Übungsumgebung zur Fouriertransformation

plikationen statt. Schließlich werde ungerade und zusammengesetzte Funktionen in der höchsten Schwierigkeitsstufe getestet.

Die diskrete Kosinustransformation

Die diskrete Kosinustransformation ist insbesondere für die Bildverarbeitung von Bedeutung. Hierzu existierte bereits eine interaktive Animation in der Form einer Studienarbeit von Holger Wons. Diese wurde im Rahmen der Dissertation von Claudia Schremmer [Sch01] bezüglich der Lernwirksamkeit bereits untersucht. Das Applet stellte jedoch damals kein CATS-Übungsapplet dar, sondern war als interaktive Simulationsumgebung gedacht. Im Rahmen einer Bachelorarbeit wurde diese Animation soweit erweitert, dass hieraus mit nur sehr geringem Aufwand ein CATS-Übungsapplet entstand. Die Studierenden erhalten nun von dem Applet immer komplizierter werdende Aufgabenstellungen. Sie müssen nun die einzelnen Frequenzkoeffizienten so einstellen, dass eine möglichst gute Rekonstruktion des vorgegebenen Signals erreicht wird. Die Toleranzwerte, innerhalb derer eine eingestellte Lösung als richtig gilt, können die Studierenden zwar selbst vorgeben, ein hoher Toleranzwert führt allerdings auch zu einer Verminderung des Schwierigkeitsgrades. Die Abbildung 7.12 stellt das Applet dar. Es sind im mittleren Bereich die Vorgabefunktion (in blauer Farbe) und Rekonstruktionskurve (in roter Farbe) sowie die Frequenzkoeffizienten zu sehen.

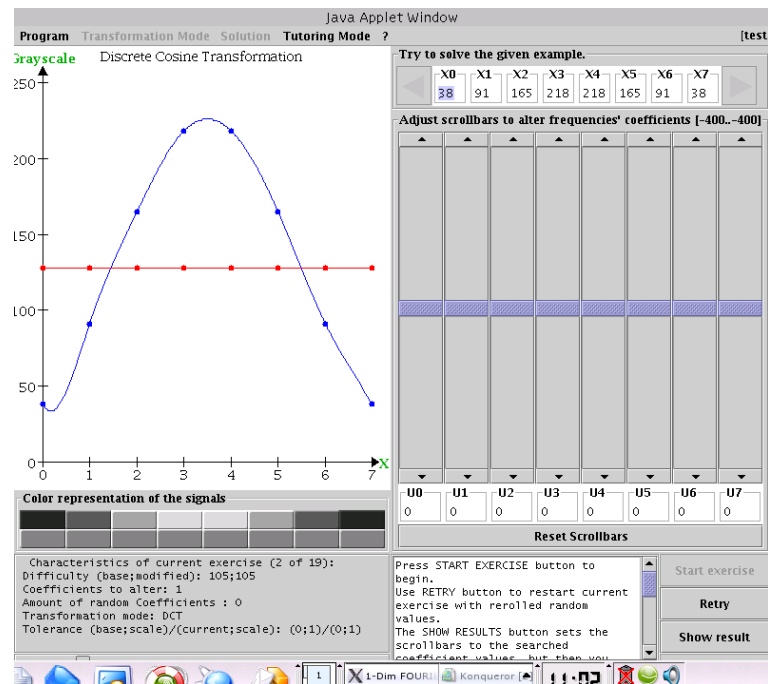


Abbildung 7.12: Die Übungsumgebung zur diskreten Kosinustransformation

Traditionelles Platten-Scheduling

Die meisten traditionellen Sekundärspeichersysteme, wie sie etwa in Medienserver verwendet werden, verwenden je nach Anforderung bestimmte Scheduling-Algorithmen. Hierbei sind z. B. folgende Anforderungen möglich: Echtzeitfähigkeit, Zuverlässigkeit, mittlere Zugriffszeit. Daher erlernen die Studierenden innerhalb der Vorlesung verschiedene traditionelle Platten-Scheduling-Verfahren. Diese Verfahren können mit Hilfe eines Applets von Katrin Heineken vertieft werden [Hei02]. Bei diesem Applet wurde wieder sehr stark der Spieltrieb der Lernenden angesprochen. Nach Absolvierung der Aufgabe stellt das Applet zudem eine animierte Lösung dar. Dadurch sollen die Studierenden ein besseres Verständnis im Bereich der Medienserver erhalten. Es werden die folgenden Zugriffverfahren trainiert:

FCFS First Come First Serve: Die einzelnen Anforderungen werden schrittweise durch den Plattentreiber akzeptiert und in der Reihenfolge ihres Eintreffens abgearbeitet. Der Algorithmus ist fair (insbesondere ein "Aushungern" einer Anforderung ist ausgeschlossen) und einfach zu implementieren. Allerdings arbeitet der Algorithmus in Bezug auf die Kopfbewegung nicht optimal, da andere Anforderungen (die z. B. auf dem Weg liegen) nicht beachtet werden.

Daraus folgt eine im Verhältnis zu den anderen Algorithmen eine sehr hohe mittlere Zugriffszeit.

SSTF Shortest Seek Time First: Dieser Algorithmus sucht, ausgehend von der aktuellen Position des Kopfes, diejenige Anforderung mit der geringsten Suchzeit. Dieses Verfahren minimiert die Zugriffszeiten und arbeitet hierfür optimal. Allerdings werden einzelne Anforderungen, solche die sich Blöcke in der Mitte der Festplatte beziehen, zuungunsten der am inneren und äußeren Rand liegenden bearbeitet.[Ste00][S. 359]

SCAN Die anfallenden Anforderungen werden zwar in der Reihenfolge des kleinsten Abstandes abgearbeitet, allerdings nur in die jeweilige Bewegungsrichtung des Schreib-/Lesekopfes; falls keine Anforderungen in dieser Richtung mehr vorhanden sind, wird die Positionier-Richtung gewechselt. Dieser Algorithmus vermeidet die Benachteiligung der mittleren Blöcke, wie beim SSTF-Algorithmus. Die Randbereiche werden besser bedient.

C-Scan Circular-Scan: Der Schreib-/Lesekopf arbeitet alle Anforderungen in der Laufrichtung bis zum Festplattenende ab. Anschließend beginnt er wieder am Anfang. Dieser Algorithmus bietet sowohl einen fairen Zugriff wie auch gleichmäßigere Wartezeiten an. Die Leistung ist etwas geringer als die des SCAN-Algorithmus

Fehlerkorrekturverfahren bei optischen Speichern

Ein mathematisch sehr anspruchsvolles Verfahren ist die Fehlerkorrektur auf Datenträgern wie der Compact Disc (CD) oder der Digital Versatile Disc (DVD). Hier sind umfangreiche Kenntnisse aus der Kodierungstheorie und den damit verbundenen mathematischen Grundlagen erforderlich. Zudem ist der Ablauf der Fehlerkorrektur bei CDs aufgrund der Mischung sehr unterschiedlicher Verfahren sehr komplex. In einer gemeinsamen Studienarbeit von Barbara Herzog [Her03b] und Manuela Mních [Mni03] konnte dieses Problem mit Hilfe einer besonderen didaktischen Begleitung gelöst werden. Zunächst erhalten die Studierenden eine genaue Übersicht über die verschiedenen Normen und die damit verbundenen Fehlerkorrekturverfahren. Hierdurch erhalten sie einen groben Überblick. Abbildung 7.13 stellt diese erste Orientierung dar.

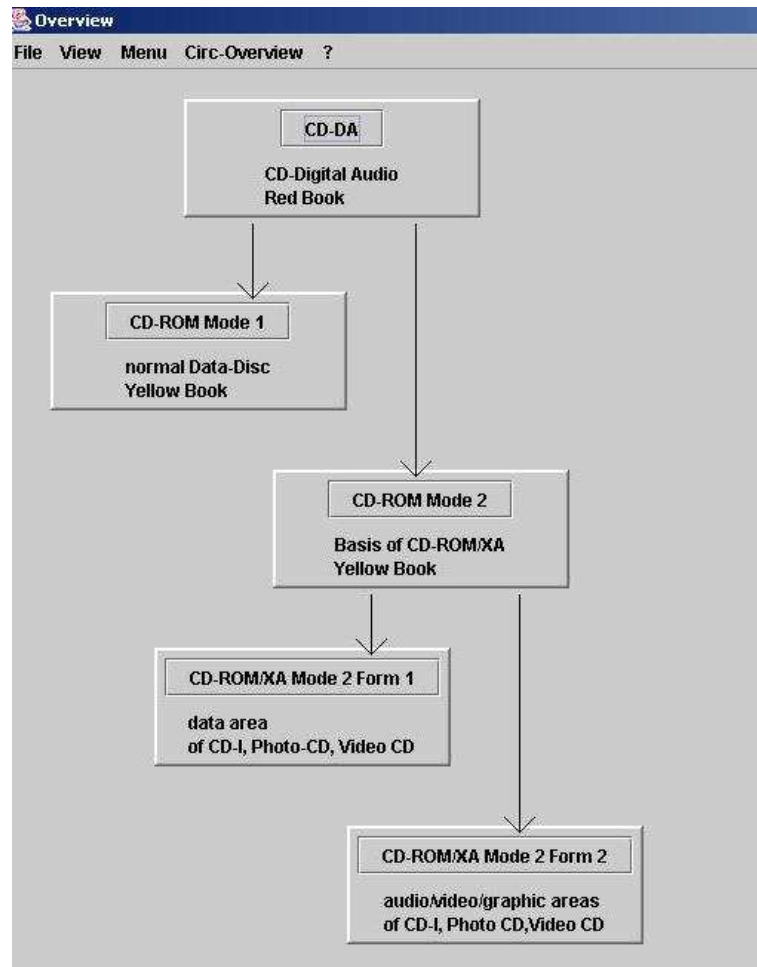


Abbildung 7.13: Navigation über die einzelnen Verfahren

In einer zweiten Navigationsebene, hier insbesondere beim Reed-Salomon-Product-Code, erhalten die Studierenden begleitend zu ihren Aufgaben eine erweiterte Navigation, die ihren aktuellen Stand im Verfahren visualisiert. Dies geschieht durch eine einfache, aber wirkungsvolle Darstellung im entsprechenden Blockschaltbild. In der Abbildung 7.14 ist diese Art der Navigation dargestellt. Der ansteigende Schwierigkeitsgrad bei diesem Applet wird dadurch erreicht, dass in den Datenströmen zunächst sehr viele 0-Bits-Reihen generiert werden. Dadurch erleichtert sich die Berechnung wesentlich. Bei schwierigeren Stufen werden dann komplexere Bitmuster zugelassen.

7 Die Evaluierung des CATS-Systems

The screenshot displays two Java applets side-by-side. The main applet, titled 'Delay', contains an 'Exercise' section with the instruction: 'Arrange the four information signals so that the CIRC-Encoder can process them!'. It lists four signals: 1. 11010011, 2. 00010010, 3. 10100101, and 4. 01101111. A 'New' button is located to the right. Below is a 'Solution' section with eight dropdown menus, each followed by a green checkmark, containing the following bit sequences: 1101, 0011, 0001, 0010, 1010, 0101, 0110, and 1111. 'Test' and 'Solution' buttons are at the bottom of this section. An 'Online' section at the very bottom has a 'Send' button and displays 'difficulty=2 valuation=1 time=69'. Navigation buttons 'Back', 'Next', 'Menu', and 'CIRC-Overview' are at the bottom of the applet. The 'CIRC-Overview' applet shows a block diagram of the CIRC encoder. It includes a 'Delay of 2 frames' block, a 'Cross-Interleaving' section with 'Q-parity' and 'P-parity' paths, and another 'Delay of 2 frames' block. The diagram is labeled with various numbers (8, 16, 24, 28, 32) and letters (C2, C1, D).

Abbildung 7.14: Navigation im Ablauf des Verfahrens

Zusammenfassung

Mit den implementierten CATS-Applets für den Bereich der Informatik konnten die oben vorgestellten Szenarien eins und zwei realisiert werden (s. Abschnitt 5.2 und 5.3). Eine Integration von Fernlernern anderer Universitäten mit Präsenzlernern war dadurch möglich. Eine Isolation der Lerner konnte durch die Möglichkeit einer Gruppenkommunikation vermindert werden. Der Schwerpunkt lag, durch Projektvorgaben beeinflusst, auf einer möglichst breiten inhaltlichen Abdeckung der Fächer Rechnernetze und Multimediatechnik. Daher standen entsprechende Ressourcen für eine rollenbasierte Gruppenarbeit, wie sie in Szenario 3 vorgestellt wurde, im Fach Informatik nicht zur Verfügung.

7.2 Beispiele aus dem Bereich der empirischen Sozialforschung

Um zu zeigen, dass CATS auch in anderen Fächern eingesetzt werden kann, wurde eine informelle Kooperation mit dem Fachbereich der Politikwissenschaft durchgeführt. Im Rahmen dieser Kooperation konnten zwei Themenbereiche mit CATS-Applets angereichert werden. Diese wurden im Rahmen des Projektes "Politikon" (Politikwissenschaft online) in die entsprechenden Lerneinheiten integriert. Das Projekt ist ein durch das Bundesministerium für Bildung und Forschung gefördertes Projekt zur Nutzung der Neuen Medien in der Hochschullehre. Es entstand in Zusammenarbeit mit dem Fachverband der Politikwissenschaft (DVPW). Die Inhalte bei diesem Projekt wurden arbeitsteilig von den Teilnehmern, die gleichzeitig auch die Teildisziplinen des Projektes repräsentieren, erstellt. Die Applets hatten im Rahmen der Politikwissenschaftlichen Methoden einerseits die anonyme Erfassung von kritischen Fragen und andererseits die statistische Auswertung nicht repräsentativen Stichproben zum Gegenstand.

7.2.1 Übungsapplet zur Randomized Response Technik (RRT)

Da die Befragung in der empirischen Sozialforschung immer noch die häufigste Art der Datenerhebung ist, kommt der möglichst wahrheitsgemäßen Beantwortung von Fragen eine hohe Bedeutung zu. Dies ist jedoch umso unwahrscheinlicher, je kritischer die Fragen (z. B. intime Fragen oder Fragen im Zusammenhang mit Straftaten) sind. Insbesondere für solche Fragen ist die Sicherstellung und Nachvollziehbarkeit der absoluten Anonymität für die Befragten von hoher Bedeutung. Aus diesem Grunde wurde die Randomized Response Technik (RRT) entwickelt.

Die Idee besteht darin, mit Hilfe einer Zufallsgröße die Anonymisierung von Antworten durchzuführen. Hierbei werden zwei Fragen miteinander verknüpft: Eine harmlose, wie z. B. "Hat Ihr Vater an einem geraden Tag Geburtstag?" mit einer kritischen Frage, wie z. B. "Haben Sie in den letzten zehn Jahren geklaut?". Anhand einer Zufallsgröße (z. B. Münzwurf), deren Ergebnis nur die Befragten kennen, wird bestimmt, welche Frage die Probanden wahrheitsgemäß zu beantworten haben. Da nur die Probanden selbst wissen, welche Frage sie beantworten, bleibt die Anonymität im Falle der kritischen Frage gewahrt. Anhand von statistischen Methoden kann nun aus dem Gesamtergebnis abgeschätzt werden, wieviel Prozent der Teilnehmer

7 Die Evaluierung des CATS-Systems

die kritische Frage bejaht haben. Die Studienarbeit von Rong Liu [Liu03] hatte einen entsprechenden Aufgabentrainer zum Gegenstand. Hierbei wurde ein entsprechendes Experiment simuliert, und die Studierenden haben dann die jeweiligen Antworten auszuwerten und den Prozentsatz der kritischen Bejahung abzuschätzen.

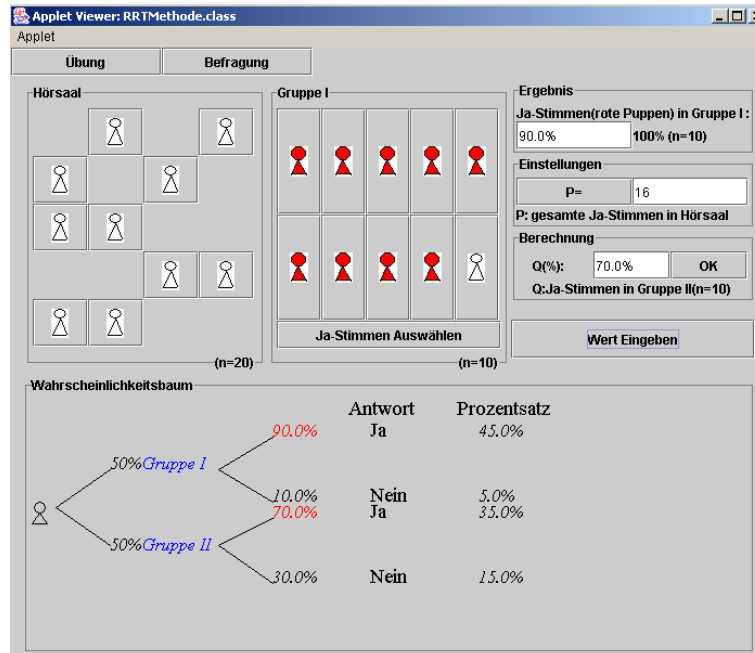


Abbildung 7.15: CATS-Applet zur Übung der Randomized Response Technik

7.2.2 CATS-Applet zum Non-Equivalent-Group-Design

In der Politikwissenschaft spielt die Erkennung von Zusammenhängen bei einer statistischen Erhebung eine wichtige Rolle (z. B. der Einfluss des Alters der Befragten auf ihr Wahlverhalten). Hierzu führt man in aller Regel so genannte repräsentative Umfragen durch. Diese beruhen auf einer weitestgehend repräsentativen Stichprobe. Es gibt jedoch auch Fallkonstellationen, in denen man keine repräsentative Stichprobe wählen kann. Hier gibt es jedoch mit Hilfe des so genannten non-equivalent-group-design die Möglichkeit, trotzdem Rückschlüsse ausgehend von einer nichtrepräsentativen Stichprobe auf die Grundgesamtheit zu erhalten. Die Verfahren sind jedoch sehr kompliziert. Daher wurde ein entsprechendes Trainingsapplet von Norbert Wirges innerhalb seiner Studienarbeit realisiert [Wir03]. Abbildung 7.16 stellt das Applet dar. Ausgehend von einer statistischen Erhebung wird eine Frage an

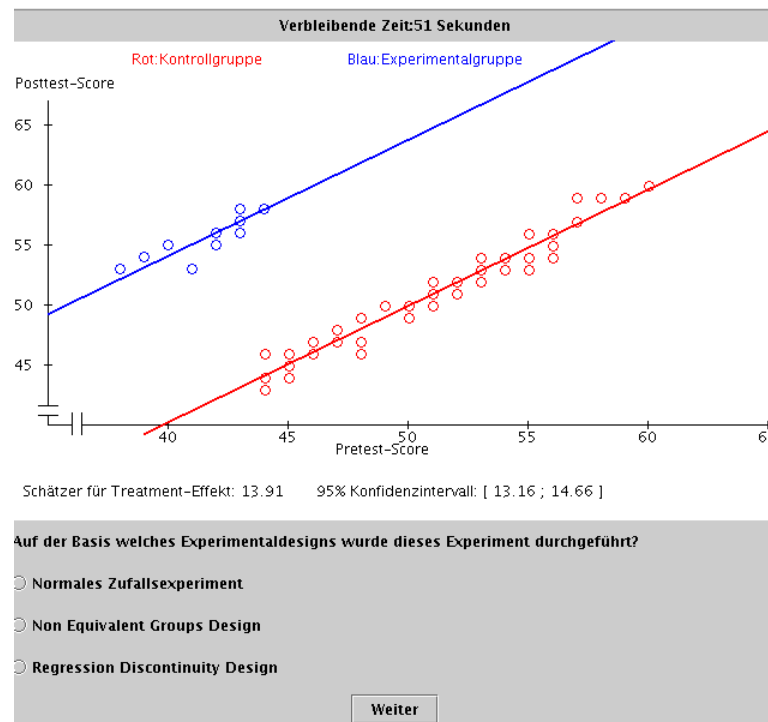


Abbildung 7.16: Das Non-Equivalent-Group-Design-Applet

die Studierenden gestellt, die abprüft, ob die Thematik des Non-Equivalent-Group-Designs ausreichend verstanden wurde. Daneben wurde ebenfalls eine statistische Simulation erstellt, mit der die Studierenden durch die Veränderung bestimmter Rahmenparameter (Anzahl der Messwerte, Validität etc.) eine mögliche Realisierung darstellen.

7.3 Die simulierte Gerichtsverhandlung

Im Rahmen dieser Arbeit entstand als Prototyp ein Applet zur Gruppenarbeit in Form einer simulierten Gerichtsverhandlung (so genannter Moot-Court). Die Idee entstand bereits im 16. Jahrhundert in England. Dort trafen sich die Studierenden der Rechtswissenschaft in Kneipen, um in Anwesenheit von erfahrenen Juristen aktuelle Fälle zu diskutieren und ihre juristische Argumentation zu üben. Dadurch war es für die Studierenden möglich, reale Fälle zu bearbeiten, ohne die Interessen echter Mandanten zu gefährden. Im Laufe der Zeit hat sich dieses Verfahren in der Ausbildung etabliert und gehört inzwischen insbesondere in angelsächsischen Ländern zum fes-

ten Bestandteil der juristischen Ausbildung. Es werden auch lokale, nationale wie internationale Wettbewerbe durchgeführt [Com04].

Dieses reale Modell wurde als Vorlage für die Anwendung der CATS-Gruppenarbeit verwendet. Hierbei wurden vier Rollen, die auch im Moot-Court vorkommen, vorgesehen: Ankläger, Verteidiger, Richter und Prozessbeobachter. Diese Art der Gruppenarbeit ist rein textbasiert. In späteren Erweiterungen können jedoch auch beliebige multimediale Elemente (z. B. Bilder, Filmsequenzen etc.) hinzugefügt werden. Als rechtliche Wissensdomäne wurde das deutsche Strafrecht ausgewählt. Ausgehend von einem vorgegebenen Schwierigkeitsgrad wird dynamisch durch das Gruppenobjekt ein konkreter Fall konstruiert. Zum Beispiel wird in der einfachsten Schwierigkeitsstufe der Fall so gehalten, dass die vier Voraussetzungen der Strafbarkeit ohne weitere Unterprüfungen abgehandelt werden können. Diese sind im Einzelnen:

Objektive Tatbestandsmerkmale Hier wird die Kausalität des menschlichen Verhaltens zu einem strafbaren Erfolg geprüft. Zum Beispiel: A schießt auf B, B ist tot. Durch die Handlung des A (schießen) ist B zu Tode gekommen. Die Kausalität menschliches Verhaltens kann bejaht werden.

Subjektive Tatbestandsmerkmale Dieses Tatbestandsmerkmal setzt bei Vorsatzdelikten das Wissen und Wollen einer menschlichen Handlung mit dem Ziel einer objektiven Tatbestandsverwirklichung voraus. Bei Fahrlässigkeitsdelikten muss stattdessen eine Pflichtverletzung gegeben sein. Beispiel: A meint in B ein Tier zu sehen, er erschießt daraufhin B. A wollte nicht B töten, sondern nur ein Tier. Ein Vorsatz ist damit nicht gegeben.

Rechtswidrigkeit Die bis dahin rechtswidrige Handlung darf nicht ausnahmsweise gerechtfertigt sein (z. B. durch Notwehr).

Schuld Schließlich könnte der Täter in ganz bestimmten Fällen noch "entschuldigt" sein, z. B. durch eine starke geistige Störung.

Bei höheren Schwierigkeitsstufen treten noch weitere Sachverhaltsergänzungen hinzu, die die rechtliche Behandlung komplizierter, da je nach Schwierigkeitsgrad in einer oder mehreren Unterprüfungen tiefergehende Analysen durchgeführt werden müssen, gestalten. Im Ausgangsfall würden z. B. noch Entschuldigungsgründe hinzugefügt werden, die aus einem möglichen Pool von Sachverhaltsergänzungen entnommen werden (z. B. Trunkenheit, Drogenrausch etc.).

Es ist zu bemerken, dass in dieser Version keine automatische Korrektur der Ergebnisse stattfindet, sondern dass aus der Bewertung durch die Rolle des Richters und durch die Reviewer eine implizite Bewertung der Gruppenleistung stattfindet.

7.4 Evaluierung in verschiedenen Lehr-/Lernszenarien

CATS wurde in verschiedenen Lehr-/Lernszenarien erfolgreich eingesetzt. Diese reichen von der Unterstützung der Präsenzlehre bis hin zum vollständigen Ersatz der Präsenzübung in der Fernlehre.

Die *Unterstützung der Präsenzlehre* fand im Rahmen der Vorlesungen und Übungen im Bereich der Rechnernetze und Multimediatechnik statt. CATS-Applets wurden während der Veranstaltung zu Demonstrationszwecken eingesetzt. Der Einsatz von Animation-Applets in Vorlesungen wird zwar bereits seit einer längeren Zeit praktiziert, doch erst durch die CATS-Applets konnte auch der Übungsbetrieb entsprechend ergänzt werden. Hierdurch wird das Handlungswissen der Studierenden mobilisiert. Dies wurde im Rahmen der Präsenzübung im SS 2004 im Fach Rechnernetze ausprobiert. Während der Übung wurden einige CATS-Applets demonstriert, was deutlich die Mitarbeit der Studierenden in der Übung erhöhte.

So genanntes *Blended Learning* ist die Kombination von Elementen der Präsenzlehre mit denen der Fernlehre. Im Rahmen dieser Arbeit wurde CATS auch in diesem Bereich eingesetzt: Den Studierenden wurde neben den klassischen Übungsaufgaben (im Fach Rechnernetze) auch aufgegeben, einzelne CATS-Aufgaben (Codierung von Bitströmen auf Leitungen, Token Ring Applet, Multicast Applet, Congestion Control Applet) selbständig zu Hause zu bearbeiten. Die Studierenden konnten dann von zu Hause aus auf das System zugreifen und beliebig oft den Stoff bearbeiten. Von dieser Möglichkeit wurde insbesondere kurz vor der Abschlussklausur rege Gebrauch gemacht.

Die *Unterstützung der Fernlehre* fand im Rahmen von zwei reinen virtuellen Fernlehrkursen des Lehrstuhl Praktische Informatik IV statt. Hierbei konnte das CATS-System den gesamten Übungsbetrieb ersetzen. Dies betraf die Vorlesungen Multimediatechnik und Rechnernetze in den beiden Projekten Universitärer Lehrverbund Informatik (ULI) und Wirtschaftsinformatik Online (Winfoline). Das CATS-System befindet sich in den angebotenen Kursen im Regelbetrieb.

Daneben sind die beiden CATS-Applets aus dem Bereich der Politikwissenschaft im Rahmen des Projektes Politikwissenschaft Online (Politikon) eingesetzt worden.

Zusammenfassend konnte durch die oben genannten Beispiele aus unterschiedlichen Wissensgebieten die Funktionsfähigkeit des CATS-System in der Lehre nachgewiesen werden (Proof of Concept).

Um auch die Akzeptanz und eine mögliche Beeinflussung des Lernerfolgs bei den Studierenden festzustellen, wurde eine entsprechende Evaluierung durchgeführt. Im Rahmen der zur Verfügung stehenden Ressourcen wurden sowohl ein Akzeptanztest wie auch eine Untersuchung des Lernerfolgs (repräsentiert durch die Klausurergebnisse) in Abhängigkeit von der Nutzung des CATS-Systems durchgeführt.

7.4.1 Rahmenbedingungen des Experiments

Während des Wintersemesters 2003/04 wurde ein Experiment bezüglich der Akzeptanz des CATS-Systems und der Examensergebnisse der CATS-Nutzer durchgeführt. Als Veranstaltung wurde die Vorlesung und Übung im Fach "Multimediatechnik" ausgewählt. Diese Veranstaltung ist ein Wahlfach für die Studierenden in den Fächern "Wirtschaftsinformatik", "Technische Informatik" und "Bachelor of Software and Internet-Technology". Es werden pro Woche je zwei Vorlesungen von 90 Minuten Dauer und eine Übung im Umfang von 90 Minuten gehalten. Alle Vorlesungen (inkl. der entsprechenden Annotationen) und Folien werden zusätzlich aufgezeichnet und über das WWW den Studierenden frei zur Verfügung gestellt. Daneben werden eine Reihe von Java-Animationen als vertiefendes Material ebenfalls über das Netz frei angeboten. Die Nutzung des CATS-Systems war im Rahmen des Experimentes freiwillig.

7.4.2 Hypothesen

Das Ziel des Experimentes war die Evaluierung folgender Hypothesen:

1. Akzeptieren die Studierenden, dass CATS ihnen eine Rückmeldung über ihren Lernstand gibt?
2. Sind die Studierenden davon überzeugt, dass sie mit dem CATS-Applet besser lernen können?

3. Existiert eine Korrelation zwischen den Examensergebnissen und der Benutzung von CATS?

7.4.3 Experiment-Design

Zu Beginn des Semesters wurde im Rahmen eines Vortests das Vorwissen über das Fachgebiet von allen Studierenden über das gesamte, in der Vorlesung zur vermittelnde Stoffgebiet abgefragt. Zu diesem Zweck wurde zunächst jedes Kapitel durch drei Multiple-Choice-Fragen repräsentiert, die in drei unterschiedlichen Schwierigkeitsstufen gestaltet wurden. Um die Rate-Wahrscheinlichkeit zu minimieren, wurde einerseits noch das Feld "weiss nicht" eingefügt und andererseits eine Frage erst dann als korrekt bewertet, wenn alle und nur die richtigen Antworten angekreuzt wurden (n aus 5).

Um systematische Fehler und Unklarheiten zu vermeiden, wurde vor dem Test der gesamte Fragebogen durch eine Expertenbefragung überprüft. Der Fragebogen wurde insgesamt sieben Personen, Kollegen des Lehrstuhls, die das Fachgebiet genau kannten, zur Stellungnahme vorgelegt. Zudem wurde auch noch eine Befragung an einem Nichtexperten durchgeführt, um insbesondere zu verhindern, dass die richtige Antwort sich bereits aus der Frage ergibt. Danach wurde der Fragebogen noch einmal gründlich überarbeitet. Insgesamt enthielt der Fragebogen 27 Fragen über die neun Kapitel des Faches "Multimediatechnik".

Die Fragebögen wurden an alle Studierende der Vorlesung und der Übung verteilt. Mit den Fragebögen wurde gleichzeitig der Link zu dem CATS-System wie auch eine kurze Gebrauchsanleitung mitgeteilt. Die Benutzung des CATS-Systems war den Studierenden freigestellt.

Nach Abschluss des Semesters wurde ein Nach-Wissenstest mit den gleichen Fachfragen durchgeführt. Dieser zweite Fragebogen enthielt allerdings noch zusätzliche Fragen über die Akzeptanz, Geschlecht und Nutzungsverhalten. Dieser Nach-Wissenstest wurde in der letzten Vorlesungsstunde vor der Klausur durchgeführt.

7.4.4 Ergebnisse des Experimentes

Die Ergebnisse des Experimentes lassen sich in zwei Teile untergliedern: In den Akzeptanztest, welcher zum Teil weitere Informationen über das Kommunikationsverhalten der Studierenden enthält, und in einen Teil, der die Nutzung des Systems

und das Lernsetting der Studierenden mit den Ergebnissen der Abschlussprüfung in Beziehung setzt.

Ergebnisse des Akzeptanz-Test

Insgesamt stellte sich heraus, dass 42 % der Studierenden das CATS-System während ihres Studiums verwendet hatten. Wie bereits erwähnt, handelte es sich um ein freiwilliges, zusätzliches Angebot. Das CATS-System stellte somit eine Ergänzung zum bestehenden Angebot dar. Die Studierenden, die das CATS-System benutzten, verwendeten es im Schnitt 41 Minuten pro Woche. Wobei die meisten Studierenden das System am späten Nachmittag bzw. frühen Abend (17:00 bis 20:00) benutzten. Tabelle 7.1 zeigt die weiteren Akzeptanzergebnisse sowie die Ergebnisse der Abfrage des Kommunikationsverhaltens.

Antwort:	Ja	Nein
Glauben Sie, dass Sie durch CATS den Inhalt besser verstanden haben?	42,9 %	57,1 %
Werden Sie CATS in der Zukunft weiterverwenden?	71,4 %	28,6 %
Benutzen Sie ein Instant-Messaging-System?	64,3 %	35,7 %
Würden Sie eine Integration von CATS in das Instant Messaging System begrüßen?	64 %	36 %

Tabelle 7.1: Ergebnisse des Akzeptanztests

Es erscheint zunächst erstaunlich, dass lediglich 42 % der Studierenden behaupten, den Lernstoff nach der Bearbeitung mit CATS besser verstanden zu haben. Hierzu ist zu bemerken, dass die Bedienung der Übungsapplets schon ein hohes Verständnis der jeweiligen Materie voraussetzt, welches in den frühen Lernphasen vermittelt wird. Die CATS-Übungsaufgaben dienen dann nur noch der Vertiefung, Reflexion und der Vermittlung von Kommunikationskontakten.

Wie sich aus der Tabelle ablesen lässt, verwenden bereits heute sehr viele Studierende Instant-Messaging-Dienste, wie ICQ oder den Microsoft Messenger. Das zeigt, dass die Kommunikationstechnik, wie sie CATS zugrunde liegt, keine nennenswerte Eintrittsbarriere darstellt. Im Gegenteil: Eine Integration in die bestehenden Kommunikationssysteme wird gerade von der deutlichen Mehrheit der CATS-Benutzer gewünscht. So ist die Hürde, dann auch die Audio- und Videokonferenzmöglichkeit des CATS-Systems zu benutzen deutlich geringer. Zurzeit sind zudem immer mehr

Instant-Messenger-Systeme dabei, in neuen Programmversionen auch Audio- und Videokonfernzmöglichkeiten zu schaffen. Dies sind z. B. der Microsoft Messenger, ICQ und Yahoo.

Dass über 70 % der CATS-Benutzer das System in der Zukunft weiterverwenden, ist ein sehr hoher Akzeptanzwert, der die Akzeptanz von anderen Übungssystemen an Fernuniversitäten deutlich übertrifft. An der FernUniversität in Hagen etwa nutzen nur ca. 10 % der Studierenden überhaupt den Lernraum Virtuelle Universität, der auch das Übungssystem Webassign beinhaltet [Fri04b].

Klausur-Ergebnisse

Eine weitere Fragestellung betraf die Korrelation zwischen der CATS-Benutzung und dem Examensergebnis. Hierbei zeigt sich ein interessantes Phänomen: Zu Beginn des Semesters besuchten noch 49 Studierende aktiv die Vorlesung und/oder die Übung. Am Ende des Semesters besuchten nur noch 31 Studierende diese beiden Veranstaltungen, es absolvierten jedoch 71 Studierende die Abschlussklausur. Der Fokus dieses Experimentes lag jedoch in einer Unterstützung der klassischen Lehre und ob sich das System als Ergänzung des Studienangebotes förderlich auf den Lernerfolg der Studierenden auswirkt.

Aufgrund der Erhebung¹ bei allen Präsenzstudierenden, der Registrierung im CATS-System und den Daten aus der Abschlussklausur lassen sich die Studierenden an der Präsenzuniversität in drei Gruppen einteilen: Die Studierenden, die im so genannten Off-Campus-Modus studieren, das heißt, sie eignen sich den Stoff ohne jegliche Betreuung zu Hause selbst an. Aufgrund von Befragungen während der Klausureinsicht oder bei mündlichen Prüfungen ist davon auszugehen, dass diese Studierenden das Angebot der Vorlesungsaufzeichnungen annehmen und sich in Heimarbeit sozusagen als Fernstudierende den Inhalt der Veranstaltung selbst vermitteln. Sie hatten allerdings keinen Zugriff auf das CATS-System. Die zweite Gruppe besucht regelmäßig die Präsenzveranstaltungen, sie benutzt jedoch nicht das CATS-System. Die dritte Gruppe schließlich besucht regelmäßig die angebotenen Präsenzveranstaltungen und nutzt zusätzlich das CATS-System.

Die Examensergebnisse wurden folgenden Gruppen zugeordnet: Die Fernstudierenden dieses Semesters sind in der Statistik nicht berücksichtigt, da nur zwei dieser Teilnehmer die Klausur mitschrieben. Die beiden Kandidaten erhielten überdurch-

¹Die Zuordnung erfolgte unter Wahrung der Anonymität der Studierenden

Gruppe der Studierenden	Anzahl	ϕ Note	Standardabw.
Alle Studierenden	71	3,4	1,2
Gruppe 1 (Off-Campus Lerner ohne Tutoring und ohne CATS)	40	3,5	1,2
Gruppe 2 (Reine Campus-Lerner ohne CATS)	19	2,95	1,15
Gruppe 3 (Blended Learner und CATS-Benutzer)	12	2,63	0,99

Tabelle 7.2: Klausurergebnisse nach CATS-Gruppen im WS 2003/04

schnittliche Noten (2,0 und 2,3). Eine qualifizierte statistische Betrachtung ist allerdings aufgrund der geringen Anzahl nicht möglich.

7.4.5 Examensergebnisse im Fach Rechnernetze aus dem SS 2004

Im Sommersemester 2004 fand die Vorlesung Rechnernetze statt. Zwar wurde keine empirische Studie durchgeführt, allerdings wurden die Examensergebnisse der unterscheidbaren Gruppen festgehalten. Hierbei wurden die Studierenden der Universität Mannheim mit den Fernstudierenden, die durch geeignete Maßnahmen aufgefordert wurden das CATS-System zu verwenden, verglichen. Der Notendurchschnitt lag bei den 82 Mannheimer Studierenden bei 3,52 mit einer Standardabweichung von 1,06 und bei den 15 Fernstudierenden (nur die, die auch die gleiche Klausur mitgeschrieben haben)² lag der Notenschnitt bei 3,16 mit einer Standardabweichung von 0,88.

Das Ergebnis überrascht, da die Fernstudierenden tendenziell ungünstigere Lernbedingungen hatten als die Präsenzstudierenden. Dies hat sich aber nicht negativ ausgewirkt, im Gegenteil: Die Fernstudierenden erreichten bessere Ergebnisse. Somit kann zumindest festgestellt werden, dass das Fernstudium und insbesondere die vollständige Ersetzung des Übungsbetriebes im Fernlehrszenario durch CATS keine negativen Auswirkungen auf den Lernerfolg der Studierenden hat.

²Die Ergebnisse der mündlichen Prüfung waren noch einmal deutlich besser, wurden aufgrund der unterschiedlichen Prüfungsverfahren in dieser Auswertung nicht berücksichtigt.

7.4.6 Diskussion der Ergebnisse der empirischen Evaluierung

Aus den Ergebnissen kann geschlossen werden, dass die Studierenden das CATS-System grundsätzlich akzeptieren und dass die Akzeptanz noch deutlich gesteigert werden könnte, wenn eine Integration mit den gewohnten Instant-Messaging-Programme durchgeführt würde. Die Mehrheit der Studierenden, die das System verwendet haben, sind bereit, es auch in Zukunft weiter einzusetzen. Die Akzeptanzrate ist damit im Vergleich zu anderen Systemen sehr hoch, insbesondere zu dem Lehrraum Virtuelle Unviversität mit dem System Webassign, welches von der Fern-Universität in Hagen bereits seit mehreren Jahren eingesetzt wird (s. Abschnitt 3.8.5, [BHSV99]).

Zudem konnte gezeigt werden, dass die CATS-Benutzer im Vergleich zu allen anderen Gruppen ein deutlich besseres Ergebnis in der Abschlussklausur zeigten, bei gleichzeitiger Reduktion der Standardabweichung. Aus den Vorwissenstests ergaben sich keine signifikanten Unterschiede in den Gruppen 2 und 3. Eine detaillierte Wissenszuwachsanalyse war jedoch aufgrund der geringen Anzahl der rückgemeldeten Bögen statistisch nicht möglich.

Die präsentierten Ergebnisse sind jedoch ein deutliches Indiz für die positive Lernwirksamkeit des CATS-System im Lernprozess der Studierenden. Die vollständige Ersetzung des Übungsbetriebes in der virtuellen Lehre durch CATS erscheint ohne negative Auswirkungen auf den Lernerfolg möglich.

Um jedoch eine eindeutige Kausalität zwischen der Nutzung von CATS und dem Lernzuwachs nachweisen zu können, wäre ein Experiment unter Laborbedingungen mit vier randomisierten Gruppen (mindestens jeweils mit 50 Studierende) notwendig. Dies ist jedoch aufgrund der geringen Gesamtzahl an Teilnehmern in beiden Veranstaltungen nicht möglich gewesen.

7.5 Zusammenfassendes Ergebnis der Evaluierung

Es konnte gezeigt werden, dass das CATS-System in unterschiedlichen Fächern, insbesondere aus den Bereichen Informatik, Politikwissenschaft und der Rechtswissenschaft, eingesetzt werden konnte. Hierbei sind unterschiedliche Arten von Aufgabenstellungen realisiert worden. Diese reichen von einfachen Multiple-Choice-Aufgaben bis hin zu animierten Aufgaben, die eine hohe Interaktion des Lernalers erfordern. Die Anwendung des CATS-Systems als Ersatz der Präsenzübung ist mitt-

7 Die Evaluierung des CATS-Systems

lerweile fester Bestandteil aller angebotenen Fernkurse am Lehrstuhl für Praktische Informatik IV. Das System arbeitet stabil und zuverlässig. Die Anbindung an eine Lernplattform konnte innerhalb von drei Wochen realisiert werden, und damit kann CATS leicht in den Lehrbetrieb der Universität Mannheim verankert werden.

CATS kann in verschiedenen Lehr-/Lernszenarien eingesetzt werden: Als ergänzendes Material für Präsenzveranstaltungen, zur Unterstützung von Fernstudierenden in der reinen Fernlehre und als Hilfsmittel für das so genannte Blended-Learning.

Schließlich ist CATS im Rahmen einer empirischen Studie evaluiert worden. Hierbei zeigte sich eine hohe Akzeptanz von Seiten der Studierenden und ein Indiz für eine positive Beeinflussung des Lernerfolges.

8 Zusammenfassung und Ausblick

8.1 Zusammenfassung

In der klassischen Lehre spielen Übungen eine entscheidende Rolle für den Lernerfolg der Studierenden. Insbesondere Gruppenübungen mit unterschiedlichen Rollen ermöglichen es, einen Wissensgegenstand aus unterschiedlichen Perspektiven zu betrachten und durch das Mittel der Externalisierung besser zu verstehen. Dies setzt allerdings eine direkte Kommunikation zwischen den Studierenden untereinander und den Lehrenden voraus.

Im Bereich der Fernlehre sind die Möglichkeiten der direkten Face-toFace-Kommunikation, aufgrund der räumlichen Trennung nur sehr eingeschränkt möglich. Es existieren zwar technische Lösungsmöglichkeiten wie z. B. Shared-Workspace-Systeme oder Lernplattformen mit integrierten Kommunikationsmöglichkeiten (Foren, E-Mail und Chat), aber diese Systeme lösen nur Teilaspekte. Insbesondere fehlt diesen Systemen eine Integration von adaptiven Gruppenaufgaben, wie auch eine integrierte Audio-/Videokonferenzmöglichkeit, die eine rollenbasierte Kommunikation unterstützt.

Das in dieser Arbeit vorgestellte Communication and Tutoring System (CATS) kombiniert zum einen adaptive Aufgaben mit einem internetbasierten und standardisierten Kommunikationssystem und ermöglicht dadurch die Vermittlung von Wissen zwischen den Studierenden und den Lehrenden. Bezüglich der Bedienung des Systems wurde hierbei das neue Konzept der objektzentrierten Kommunikation eingesetzt.

Zum anderen wird ein neuartiger genereller Ansatz zur technischen Abwicklung von Gruppenarbeiten in der Fernlehre vorgestellt, der auch für andere Fachbereiche einsetzbar ist. Dieser basiert auf ein Rollenkonzept, bei dem eine Rolle durch ein Bündel von Rechten ausgedrückt wird. Diese Rechte erlauben bestimmte Zugriffe auf Objekte der Gruppenarbeit. Erst durch ein zielgerichtetes Zusammenwirken und durch den damit verbundenen Wissensaustausch aller Beteiligten können ent-

sprechende Aufgaben gelöst werden.

Damit wurde Forderungen aus der pädagogischen Psychologie Rechnung getragen, die bisher, insbesondere im Bereich der Fernlehre, nicht erfüllt werden konnten.

Da CATS auch in nicht-technischen Fächern eingesetzt werden soll, wurden zwei leicht zu bedienende Autorenwerkzeuge entwickelt. Mit diesen ist es auch ohne spezielle Programmierkenntnisse möglich, CATS-Applets zu entwickeln.

Bei der technischen Umsetzung wurde auf ein hohes Mass an Flexibilität geachtet, ohne auf eine benutzerfreundliche Bedienung zu verzichten. Dies äußert sich unter anderem in der sinnvollen Modularisierung der einzelnen CATS-Komponenten. So ist es möglich, die jeweiligen Aufgaben auch offline, ohne einen besonderen Server als Übungsapplets zu verwenden. Der CATS-Server kann zudem während der Laufzeit um neue Aufgabenkomponenten erweitert werden.

Das CATS-System verfügt zur leichten Integration in verbreitete Lernplattformen über entsprechende Schnittstellen. In dieser Arbeit wurde das CATS-Integrationsmodell vorgestellt und exemplarisch die Anbindung an die Lernplattform .LRN demonstriert, die auch an der Universität Mannheim im Produktivbetrieb eingesetzt wird.

Im Kapitel Evaluierung konnte gezeigt werden, dass das System erfolgreich in Veranstaltungen der Informatik, der Politikwissenschaft und der Rechtswissenschaft eingesetzt werden kann. Ein langfristiger stabiler Betrieb ist möglich, und Online-Übungen können im Fernlehrszenario durch das CATS-System abgewickelt werden.

Durch die Auswertung einer empirischen Studie, welche den Übungsbetrieb zur Vorlesung und Übung im Fach Multimediatechnik im WS 2003/04 analysierte, konnte trotz Ergänzung eines bestehenden Präsenzangebotes eine hohe Akzeptanz des CATS-Systems bei den Studierenden wie auch ein Indiz zur Lernförderung nachgewiesen werden. Die Ergebnisse einer Auswertung des Examens im SS 2004 aus dem Bereich Rechnernetze zeigte, dass die Fernstudierenden mit CATS deutlich bessere Ergebnisse erzielten als die Präsenzstudierenden. Damit besteht keine erkennbare Benachteiligung von Fernstudierenden mehr, was insbesondere im Hinblick auf den fehlenden Face-to-Face-Übungsbetrieb befürchtet worden war.

8.2 Ausblick

Mögliche zukünftige Erweiterungen lassen sich in unterschiedlichen Bereichen vorstellen: Zum einen könnten die Gruppenapplets noch ergänzt werden, mit dem Ziel

eine komplette Veranstaltung durch Gruppenaufgaben abzubilden.

Zum anderen könnte die Kommunikationsvermittlung dahingehend erweitert werden, dass die aus den Projekt Cognitive-Tutor bekannten didaktischen Rückmeldungen durch direkte Kommunikationsverbindungen ersetzt werden. Dies würde bedeuten, dass die Studierenden in Abhängigkeit ihres Vorwissens und ihrer Lernhistorie mit den Personen verbunden werden, die genau auf die jeweiligen Lerndefizite persönlich eingehen können. Hierbei kann die Übermittlung der entsprechenden Lernhistorie an den Dozenten nützlich sein. Durch den Austausch gemeinsamer Erfahrungen zwischen den Studierenden kann dann der Lernerfolg wahrscheinlich deutlich verbessert werden.

Auch wäre eine Verbindung mit dem Projekt "WIL-MA" (Wireless Interactive Learning-Mannheim) denkbar. Basierend auf den Quizfragen in der Vorlesung könnten CATS-Übungsaufgaben zusammengestellt werden. Oder es wäre möglich, die Ergebnisse des CATS-Systems für die nächste Vorlesung durch entsprechende WIL-MA-Quiz-Fragen aufzubereiten, damit dann der Dozent auf spezielle Defizite der Gesamtheit aller Studierenden eingehen kann. Dadurch könnten Fehlentwicklungen im Lernfortschritt der Studierenden frühzeitig erkannt und beseitigt werden.

Ein weiteres Forschungsfeld wäre die Modellbildung für adaptive Gruppenaufgaben. Hier wäre es interessant, mit dem Ziel eines Aufgabeneditors für Gruppenaufgaben, sowohl ein Vorgehensmodell, eine formale Beschreibungssprache und ein geeignetes Softwarewerkzeug zu entwickeln.

Literaturverzeichnis

- [ACKP95] ANDERSON, JOHN R., A.T. CORBETT, K.R. KOEDINGER und R. PELLETIER: *Cognitive Tutors: Lessons Learned*. Journal of the Learning Science, 4:167–207, 1995.
- [Adl02] ADLER, CHRISTIANE: *Java-basiertes Trainingsystem zur Fouriertransformation*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, Dezember 2002. Studienarbeit.
- [All77] ALLEN, THOMAS JOHN: *Managing the flow of technology*. MIT Press, Cambridge, 1977.
- [All97] ALLEN, THOMAS JOHN: *Architecture and Communication: Among Product Development Engineers*. Technischer Bericht 165-97, MIT, Cambridge, September 1997.
- [Ame02] AMERICAN ONLINE INC.: <http://www.aim.com>, 2002. Letzter Besuch: 06.12.2002.
- [And83] ANDERSON, JOHN R.: *The Architecture of Cognition*. Harvard University Press, Cambridge, 1983.
- [Ang78] ANGERMEIER W. F.: *Lerntheorien*, 1978. Unterlagen der FernUniversität Hagen, Kurs 3254.
- [Aro00] ARORA, RAKESH: *Voice over ip: Protocols and standards*. Technischer Bericht, Ohio State University, Juni 2000.
- [BBC00] BRANSFORD, J.D., A.L. BROWN und R.R. COCKING: *How People Learn. Brain, Mind, Experience and School*. National Academic Press, Washington D.C., 2000.

- [BCD89] BROWN, J.S., A. COLLINS und P. DUGUID: *Situated learning and the culture of learning*. Educational Researcher, 1(18):32–41, 1989.
- [Bei03] BEIER, MARKUS: *Kommunikation in Tele-Teaching Szenarios auf Basis des Session Initiation Protocols*. Studienarbeit am Lehrstuhl für Praktische Informatik IV, Universität Mannheim, 2003.
- [BHSV99] BRUNSMANN, J., A. HOMRIGHAUSEN, H.-W. SIX und J. VOSS: *Assignments in a Virtual University -The WebAssign- System*. In: *Proceedings of the 19th World Conference on Open Learning and Distance Education*, Wien, Österreich, 1999. ICDE.
- [BKR93] BOOKSTEIN, ABRAHAM, SHMUEL T. KLEIN und TIMO RAITA: *Is Huffman coding dead? (extended abstract)*. In: *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, Seiten 80–87. ACM Press, 1993.
- [Bla02] BLACKBOARD INC. WASHINGTON D.C.: <http://www.blackboard.com>, 2002. Letzter Besuch: 04.12.2002.
- [Blo84] BLOOM, B.S.: *The 2-Sigma problem: The search for methods of group instruction as effective as one-to-one tutoring*. Educational Researcher, (13):4–26, 1984.
- [Bro85] BROWN, J.S.: *Process versus Product: A Perspective on Tools for Communcal and Informal Electronic Learning*. Journal of Educational Computing Research, 2(1):179–201, 1985.
- [Bru66] BRUNNER, JEROME: *Toward a Theory of Instruction*. Harvard University Press, Cambridge, 1966.
- [Bru04] BRUSILOVSKY, PETER: *KnowledgeTree: a distributed architecture for adaptive e-learning*. In: *WWW Alt. '04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, Seiten 104–113. ACM Press, 2004.
- [BRW03] BECKS, ANDREAS, TIM REICHLING und VOLKER WULF: *Supporting Collaborative Learning by Matching Human Actors*. In: RALPH

- H. SPRAGUE, JR. (Herausgeber): *Proceedings of the 36'th Hawaii International Conference on System Sciences (HICSS-36)*, Seiten 32–40, Big Island, Hawaii, Januar 2003. IEEE Computer Society.
- [BSH01] BECK, J., M. STERN und E. HAUGSJAA: *Applications of AI in Education*, 2001. Letzter Besuch: 03.01.2004.
- [BTL80] BERGER, P.L. und T-LUCKMANN: *Die gesellschaftliche Konstruktion der Wirklichkeit: Eine Theorie der Wissenssoziologie*. Frankfurt/Main, 1980.
- [Bul99] BULLINGER, H.-J.: *Wissen und Information als Produktionsfaktor*. Zeitschrift für wirtschaftlichen Fabrikbetrieb (ZWF), 94:83–84, 1999.
- [CBN89] COLLINS, A., J.S. BROWN und S.E. NEWMAN: *Cognitive Apprenticeship, Teaching in the Crafts of Reading, Writing and Mathematics*. In: RESNICK, L.B. (Herausgeber): *Knowing Learning and Instruction*, Seiten 453–494. Lawrence Erlbaum Associates, Hillsdale, NJ, 1989.
- [CHB01] CHUNG, G.K.W.K., T.C. HARMON und E.L. BAKER: *The impact of a simulation-based learning design project on student learning*. IEEE Transactions on Education, 44(4):390–398, November 2001.
- [Che76] CHEN, PETER PIN-SHAN: *The entity-relationship model toward a unified view of data*. ACM Trans. Database Syst., 1(1):9–36, 1976.
- [CHWS⁺02] C. HU, C., A.F.L. WONG, L. SHARPE, L. CRAWFORD, S. GOPINATHAN, M.S. KHINE und S.N. MOO: *Building a Learning Community via Videoconferencing*. In: *Proceedings of the International Conference on Computers in Education (ICCE'02)*, Seiten 411–412. IEEE Computer Society, Dezember 2002.
- [Cla98] CLAUS, VOLKER: *Informatik und Ausbildung*. Springer-Verlag, Berlin, Heidelberg, New-York, 1. Auflage, 1998.
- [Com04] COMPETITION, THE EUROPEAN LAW MOOT COURT: *The European Law Moot Court Competition*. <http://www.elmc.org/society/history.htm>, 2004. Letzter Besuch: 18.10.2004.

- [CS05] CISCO SYSTEMS, INC.: *White Paper H.323 and SIP Integration*, Februar 2005. Letzter Besuch: 22.02.2005.
- [CWG⁺02] CONIAN, OWEN, VINCENT WADE, MARK GARGAN, CORD HOCKE-MEYER und DIETRICH ALBERT: *An Architecture for Integrating Adaptive Hypermedia Services with Open Learning Environments*. In: *Proceedings of the World Conference on Educational Multimediam Hypermedia and Telecommunications (ED-MEDIA 2002, Band 2002, Seiten 344–350, Denver, Juni 2002*.
- [Des37] DESCARTES, RENE: *Discours des la me'thode*. 1637.
- [dICbdFiH04] HAGEN, GESCHÄFTSSTELLE DER INITIATIVE CAMPUSSOURCE BEI DER FERNUNIVERSITÄT IN: <http://www.campussource.de>. WWW, August 2004. Letzter Besuch: 23.08.2004.
- [DKEV00] DILLON, W.E., G.V. KONDRASKE, L.J. EVERETT und R.A. VOLZ: *Performance theory based outcome measurement in engineering education and training*. IEEE Transactions on Education, 43(2):92–99, Mai 2000.
- [Dow02] DOWLING, C.: *The socially interactive pedagogical agent within online learning communities*. In: *Proceedings of the International Conference on Computers in Education*, Seiten 30–34. IEEE Computer Society, Dezember 2002.
- [Ege03] EGENBERGER, MARKUS STEFAN: *Java Applikationsentwicklung zum Zwecke des Selbststudiums für das Shortest Path Problem im Bereich Routing*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, September 2003. Studienarbeit.
- [EGFvG03] EICHELBERGER, H., G.FISCHER, F.GRUPP und J. WOLFF V. GULDENBERG: *Programmierausbildung Online*. In: BODE, ARNDT, JÖRG DESEL, SABINE RATHMAYER und MARTIN WESSNER (Herausgeber): *DeLFI 2003: Die 1. e-Learning Fachtagung Informatik*, Band P-37 der Reihe *Lecture Notes in Informatics (LNI)*, Seiten 134–143, Bonn, September 2003. GI, Gesellschaft für Informatik (GI).

- [Eul94] EULER: *Multimediales Lernen Theoretische Fundierungen und Forschungsstand*. Unterrichtswissenschaft, 4:S. 298, 1994. Jahrgang 22.
- [FG04] FLAU, HERMANN und ROLF GESERICK: *Neue Medien in der Bildung -Hochschulen Kursbuch eLearning 2004 Produkte aus dem Förderprogramm-*. DLR-Projektträger Neue Medien in der Bildung + Fachinformation, Granthamallee 2-8, 53757 St. Augustin, im Auftrag des Bundesministeriums für Bildung und Forschung (BMBF), 2004.
- [Fri04a] FRITSCH, HELMUT: *Impact of Organisational Aspects on Drop-Out in E-Learning and Distance Education- Report of Experiences*. www.change.co.nz/docs/eden/Fritsch.pdf, 2004. Letzter Besuch: 19.10.2004.
- [Fri04b] FRITSCH, HELMUT: *Umfrage zur Nutzung der LVU*. E-Mail, 2004.
- [GAD02] GARRIS, ROSEMARY, ROBERT AHLERS und JAMES E. DRISKELL: *Games, motivation, and learning: A research and practice model*. In: *Simulation & Gaming*, Band 33, Seiten 441–467. 2002. Sage Publications.
- [Gra05] GRAUER, MICHAEL: *Entwurf und Entwicklung eines WLAN-Übungsapplets für CATS*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, März 2005. Studienarbeit.
- [GVCM03] GEYER, WERNER, JÜRGEN VOGEL, LI-TE CHENG und MICHAEL MULLER: *Supporting activity-centric collaboration through peer-to-peer shared objects*. In: *Proceedings of the 2003 international ACM SIG-GROUP conference on Supporting group work*, Seiten 115–124. ACM Press, 2003.
- [GXP01] GRAESSER, A.C., HU XIANGEN und N. PERSON: *Teaching with the help of talking heads*. In: *Proceedings of the IEEE International Conference on Advanced Learning Technologies*, Seiten 460–461. IEEE, August 2001.
- [Ham86] HAMAKER, CH.: *The Effects of Adjunct Questions on Prose Learning*. *Review of Educational Research*, 56(2):212–242, 1986.

- [Hay94] HAYNIE W. J.: *Effects of Multiple-Choice and Short-Answer Tests on Delayed Retention Learning*. The Journal of Technology Education, 6(1), 1994.
- [HB00] HOSKING, D.M. und R. BOUWEN: *Organizational learning: Relational-constructionist approaches*. European Journal of Work and Organisational Psychology, 9(2):129–132, 2000.
- [Hec89] HECKHAUSEN, H.: *Motivation und Handeln*. Springer, Berlin, 2. Auflage, 1989.
- [Hei02] HEINEKEN, KATRIN: *Traditionelles Platten-Scheduling Applet zur Übung und Erklärung FISCLI*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, September 2002. Studienarbeit.
- [Her03a] HERMANN, MARC: *Java-basiertes Tutoring-System zu dem Thema "Multicast-Routing"*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, April 2003. Studienarbeit.
- [Her03b] HERZOG, BARBARA: *Java- basierte Entwicklung einer Lernsoftwarekomponente über die grundlegende Fehlererkennung/-korrektur der Compact Disc*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, Juli 2003. Studienarbeit.
- [HKA03] HEFFERNAN, NEIL T., KENNETH R. KOEDINGER und VINCENT A. W. M. M. ALEVEN: *Tools Towards Reducing the Behavior Representation in Modelling and Simulation*. In: *Proceedings of the Conference on Behavior Representation in Modelling and Simulation (BRIMS)*, 2003.
- [HS85] HAMBLETON, R.K. und H. SWAMINATHAN: *Item response Theory: Principles and applications*. Kluwer-Nijhoff, Boston, 1985.
- [HSS⁺01] HANDLY, M., H. SCHULTZRINNE, WE. SCHOOLER, J. ROSENBERG, CAMRILLO, JOHNSTON und SPARKS: *RFC 2543bis 05 SIP: Session Initiation Protocol*. Internet Draft, 2001.

- [Huf52] HUFFMAN, DAVID A.: *A Method for the Construction of Minimum Redundancy Codes*. In: *Proceeding of the Institute of Radio Engineers*, Band 40, Seiten 1098–1101, 1952.
- [Höf01] HÖFE, OTFRIED: *Kleine Geschichte der Philosophie*. Verlag C.H. Beck, München, 2001.
- [icG] GMBH IM-C: <http://im-c.de>. Letzter Besuch: 04.12.2002.
- [IGLC04] IMS GLOBAL LEARNING CONSORTIUM, INC.: <http://www.imsglobal.org/question/index.cfm>. WWW, August 2004. Letzter Besuch: 23.08.2004.
- [Inc02] INC., ICQ: <http://www.icq.com>, 2002. Letzter Besuch: 06.12.2002.
- [Ini05] INITIATIVE, ADVANCED DISTRIBUTED LEARNING: *Advanced Distributed Learning Initiative -Homepage*. <http://www.adlnet.org>, 2005. Letzter Besuch: 11.02.2005.
- [JLSS01] JIANG, WENYU, JONATHAN LENNOX, HENNING SCHULZRINNE und KUNDAN SINGH: *Towards Junking the Pbx: Deploying IP Telephony*. In: *Proceedings of the Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Port Jefferson, New York, USA, Juni 2001.
- [Jon91] JONASSON, D.H.: *Objectivism versus Constructivism: Do we need a new philosophical paradigm?* *Educational Technology, Research & Development*, 39(3):S. 14, 1991.
- [KAH03] KOEDINGER, KENNETH R., VINCENT A.W.M.M. ALEVEN und NEIL HEFFERNAN: *Toward a Rapid Development Environment for Cognitive Tutors*. In: HOPPE, U., F. VERDEJO und J. KAY (Herausgeber): *Proceedings of the 11th International Conference on Artificial Intelligence in Education, AI-ED 2003*, Seiten 455–457, Amsterdam, 2003. IOS Press.
- [Kap02] KAPP, CHRISTINE: *Java-basiertes Tutoring-System zu dem Thema "Congestion Control in Rechnernetzen"*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, Juli 2002. Studienarbeit.

- [KH02] KURZ, GÜNTHER und HEIDE HÜBNER: *Electronic Interactive Self Assessment in the Introductory Physics Course*. In: *Proceedings of the 4th International Conference on New Educational Environment*, Seite 1.1/19, Lugano, Schweiz, 2002. SUSPI.
- [KK] KANDZIA, PAUL-THOMAS und GABRIELE KRAUS: *Universitärer Lehrverbund Informatik (ULI) Homepage*. <http://www.uli-campus.de>. Letzter Besuch: 3.12.2003.
- [Kon04] KONUROV, LIDIA: *Entwicklung und Implementierung eines Übungsapplets zur Vertiefung des Token Ring Standards*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, Februar 2004. Studienarbeit.
- [Koz87] KOZMA, R.B.: *The Implications of Cognitive Psychology für Computer-Based Learning Tools*. *Educational Technology*, 11(27):20–25, 1987.
- [KPCM02] KERN, V. M., J.M. PERNIGOTTI, M.M. CALEGARO und BENTO M.: *Peer review in engineering education: speeding up learning, looking for a paradigm shift*. In: *Proceedings of the 7th International Conference On Engineering and Technology Education- INTERTECH*, Santos, 2002.
- [Krc00] KRCMAR, HELMUT: *Informationsmanagement*. Springer, Berlin, Heidelberg, 2000.
- [Kuh01] KUHMÜNCH, CHRISTOPH: *Videoskalierung und Integration interaktiver Elemente in Teleteaching Szenarien*. Doktorarbeit, Universität Mannheim, Mannheim, Mai 2001.
- [LBJHH02] LIU, CHEN-CHUNG, BAW-JHIUNE, TZU-AN HUI und JORNG-TZONG HORNG: *Web based peer assessment using knowledge acquisition techniques: tools for supporting context awareness*. In: *Proceedings of the International Conference on Computers in Education*, Seiten 1338–1339, Dezember 2002.
- [LBW98] LEBOLD, W.K., D.D. BUDNY und S.K. WARD: *Understanding of mathematics and science: efficient models for student assessments*. *IEEE Transactions on Education*, 41(1):8–16, Februar 1998.

- [LE03a] LIEBIG, HANS CHRISTIAN und WOLFGANG EFFELSBERG: *Automatic Adaption of Exercise Problems to the Proficiency Level of the Learner*. In: *Proceedings of the 5th International Conference on New Educational Environments*, Seiten 291–296, Lucerne, Schweiz, Mai 2003.
- [LE03b] LIEBIG, HANS CHRISTIAN und WOLFGANG EFFELSBERG: *Seamless Integration of Exercise Problems to the Proficiency Level of the Learner*. In: *Proceedings of the IEEE International Workshop on Multimedia Technologies in E-Learning and Collaboration (WOMTEC)*, Nizza, Frankreich, Oktober 2003.
- [LE04a] LIEBIG, HANS CHRISTIAN und WOLFGANG EFFELSBERG: *Computer-supported Formation of Virtual Learning Groups based on Proficiency Levels*. In: CANTONI, LORENZO und CATHERINE MCLOUGHLIN (Herausgeber): *Proceedings of the ED-MEDIA 2004*, Seiten 1171–1178, P.O. Box 3728, Norfolk, VA 23514-3728 USA, Juni 2004. Association for the Advancement of Computing in Education (AACE). Ausgezeichnet mit dem Outstanding Paper Award.
- [LE04b] LIEBIG, HANS CHRISTIAN und WOLFGANG EFFELSBERG: *First Experiences with the Communication and Tutoring System (CATS)*. In: JUTZ, CHRISTINE, MATTHIAS HIRT und CORNELIA RIZEK-PFISTER (Herausgeber): *Proceedings of the 6th International Conference on New Educational Environments*, Neuchâtel, Schweiz, September 2004. net4net.
- [Liu03] LIU, RONG: *Java-basiertes Übungsapplet zur Datenerhebungsmethode "Randomized Response Methode"*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, Mai 2003. Studienarbeit.
- [LLY01] LIN, SUNNY SAN-JU, E.Z.-F. LIU und SHYAN-MING YUAN: *Web based peer assessment: attitude and achievement*. IEEE Transactions on Education, 44(2):13 pp., Mai 2001.
- [Loc90] LOCKE, JOHN: *An Essay Concerning Human Understanding*. 1690.
- [Lor77] LORD, F. M.: *A broad-range tailored test of verbal ability*. Applied Psychological Measurement, (1):95–100, 1977.

- [LPIISE01] LEHRGEBIET PRAKTISCHE INFORMATIK III SOFTWARE ENGINEERING, FERNUNIVERSITÄT IN HAGEN: *WebAssign Übungen im Internet -Benutzerhandbuch*, März 2001.
- [.LR04] .LRN BOARD: *.LRN Homepage*. <http://www.dotlrn.org>, August 2004. Letzter Besuch: 23.08.2004.
- [Ltd03] LTD, EQUIVALENCE PTY: <http://www.openh323.org>. Web-Site, Oktober 2003. Letzter Besuch: 1.10.2003.
- [Lus92] LUSTI, MARKUS: *Intelligente tutorielle Systeme*. Oldenbourg Verlag, München, 1992.
- [MBB01] MATKO, D., S. BLAZIC und A. BELIC: *Virtual race as an examination test: models, solutions, experiences*. IEEE Transactions on Education, 44(4):342–346, November 2001.
- [MCTR94] MAYES, T., L. COVENTRY, A. THOMSON und R.MASON: *Learning through Telematics: A Learning Framework for Telecommunication Applications in Higher Education*. Technischer Bericht, British Telecom, Martlesham Heath, 1994.
- [Mic02a] MICROSOFT CORPORATION: <http://messenger.msn.de>, 2002.
- [Mic02b] MICROSOFT CORPORATION: <http://www.microsoft.com/windows/net-meeting/>, 2002. Letzter Besuch: 04.12.2002.
- [Mil05] MILLIGAN, COLIN: *Reload-Web-Seite*. <http://www.reload.ac.uk/>, 2005. Letzter Besuch: 8.02.2005.
- [Mit98] MITTRACH, SILKE: *Lehren und Lernen in der virtuellen Universität*. Doktorarbeit, FernUniversität Hagen, Hagen, 1998.
- [MN99] MEIJER, ROB R. und MICHAEL L. NERING: *Computerized Adaptive Testing: Overview and Introduction*. Applied Psychological Measurement, 23(3):187–194, September 1999.
- [Mni03] MNICH, MANUELA: *Java-basierte Entwicklung einer Lernsoftwarekomponente über die erweiterte Fehlererkennung/-korrektur der Compact Disc*. <http://www.informatik.uni->

- mannheim.de/pi4/info/animation/main_animation.html, Juli 2003. Studienarbeit.
- [Mom01] MOMJIAN, BRUCE: *PostgreSQL Einführung und Konzepte*. Addison-Wesly, München, Boston, San Francisco, Harlow England, Don Mills Ontario, Sydney, Mexico City, Madrid, Amsterdam, 2001.
- [Mor95] MORRISON G. R. AND S.M. ROSS AND M. GOPALAKRISHNAN AND J. CASEY: *The effects of feedback and incentives on achievement in computer-based instruction*. Contemporary Educational Psychology, 20:32–50, 1995.
- [Mur98] MURRAY, TOM: *Authoring Knowledge Based Tutors: Tools for Content, Instructional Strategy, Student Model, and Interface Design*. Journal of the Learning Sciences, 7(1):5–64, 1998.
- [MV87] MATURANA, H. R. und F.J. VARELA: *Der Baum der Erkenntnis*. Bern, 1987.
- [Nat04] NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST): *NIST SIP - SIP parser and IP telephony support tools*. <http://snad.ncsl.nist.gov/proj/iptel/>, 2004. Letzter Besuch: 09.09.2004.
- [NT95] NONAKA, IKUJIRO und HIROTAKA TAKEUCHI: *The Knowledge-Creating Company*. Oxford University Press, New York, Oxford, 1995.
- [NT97] NONAKA, IKUJIRO und HIROTAKA TAKEUCHI: *Die Organisation des Wissens*. Campus-Verlag, Frankfurt am Main, 1997.
- [Nyk00] NYKÄNEN, OSSI: *A Framework for Generating Non-trivial Interactive Mathematical Exercises in the Web: Dynamic Exercises*. In: *World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Seiten 1473–1474, Norfolk, VA, USA, 2000. Association for the Advancement of Computing in Education (AACE).
- [Owe75] OWEN, R. J.: *A Bayesian sequential procedure for quantal response in the context of adaptive mental testing*. Journal of the American Statistical Association, (70):351–356, 1975.

- [PAS02] PESTER, ANDREAS, MICHAEL E. AUER und WOLFGANG SCHERR: *Web-Based Simulation in Virtual Lab Enviroments for Electronics*. In: *Proceedings of the 4th International Conference on New Educational Enviroment*, Seiten 361–362, Lugano, Schweiz, Mai 2002. SUSPI.
- [Paw53] PAWLOW, IWAN PETROWITSCH: *Sämtliche Werke*. Akademie-Verlag, Berlin, 1953.
- [Pet05] PETERWITZ, JULIA: *Java-basiertes, kognitives Tutoring-System der Huffman-Codierung mit adaptiven Feedback*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, Februar 2005. Bachelor-Abschlussarbeit.
- [Pia77] PIAGET, JEAN: *Epistemology and psychology of functions*. Reidel Publ., Dordbrecht, Holland, 1977.
- [PWBWYM98] PFISTER, H.-R., M. WESSNER, J. BECK-WILSON und YAND R. STEINMETZ Y. MIAO: *Rooms, protocols, and nets: Metaphors for computer supported cooperative learning of distributed groups*. In: *Proceedings of ICLS 98, International Conference of the Learning*, Seiten 242–248. Association for the Advancement of Computing in Education (AACE), Association for the Advancement of Computing in Education (AACE), 1998.
- [RS98] ROSENBERG, J. und H. SCHULTZRINNE: *A comparsion of SIP an H.323 for internet telephony*. In: *Proceedings of 1998 Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Cambridge, England, Juli 1998.
- [Rus97] RUSSELL, BERTRAND: *Denker des Abendlandes*. Gondrom Verlag GmbH, Bindlach, 1997.
- [Röd99] RÖDDER, WILHELM: *Planungs- und Entscheidungstechniken*. Kurs 0512, FernUniversität in Hagen, 1999.
- [SA89] SINGLEY, M.K. und JOHN. R. ANDERSON: *Transfer of Cognitive Skill*. Harvard University Press, Cambridge, MA, 1989.

- [Saf03] SAFAI, SASAN: *Java-basiertes Tutoring-System zu dem Thema "Leitungskodierung in Rechnernetzen"*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, 2003. Studienarbeit.
- [SCFJ96a] SCHULZRINNE, H., S. CASNER, R. FREDERICK und V. JACOBSON: *RFC 1889-RTP: A Transport Protocol for Real-Time Applications*. <http://www.ietf.org/rfc/rfc1889.txt?number=1889>, January 1996. RFC.
- [SCFJ96b] SCHULZRINNE, H., S. CASNER, R. FREDERICK und V. JACOBSON: *RFC 1889 RTP: A Transport Protocol for Real-Time Applications*. Internet Draft, Januar 1996.
- [Sch01] SCHREMMER, CLAUDIA KERSTIN: *Multimedia Applications of the Wavelet Transform*. Doktorarbeit, Universität Mannheim, Mannheim, 2001.
- [Sch03] SCHULMEISTER, ROLF: *Lernplattformen für das virtuelle Lernen: Evaluation und Didaktik*. Oldenbourg Verlag, München, Wien, 2003.
- [Sei75] SEIDNER, C.: *Teaching with Simulations and Games*. In: DUKES, R. und C. SEIDNER (Herausgeber): *Learning with Simulations and Games*, Seiten 11–45. Sage, Berverly Hills, CA, 1975.
- [SF98] SCHILL, A. und K. FRANZE: *Teleteaching: Interaktives Online-Lernen mit JaTeK - Lehrstunden auf Java*. Office Management, (6):24–25, 1998.
- [She96] SHERMAN X. HUANG: *A Content-Balanced Adaptive Testing Algorithm for Computer-Based Training Systems*. In: *Proceedings of the Third International Conference ITS 96*, Seiten 306–314, Berlin, Heidelberg, New York, 1996. Springer LNCS.
- [Ski74] SKINNER, B.F.: *About behaviorism*. Random House, Inc, New York, 1974.
- [Sla80] SLAVIN, R. E.: *Cooperative Learning*. Review of Educational Research, 50:313–342, 1980.

- [Sla93] SLAVIN, R.E.: *Synthesis of research on cooperative learning*. In: WOOLFOLK, A. E. (Herausgeber): *Readings & cases in educational psychology*, Seiten 170–178. Needham Heights: Allyn & Bacon., 1993.
- [SM94] STRAUS, S. G. und J. E. MCGRATH: *Does the medium matter? The interaction of task type and technology on group performance and member reactions*. *Journal of Applied Psychology*, (79):87–97, 1994.
- [SME⁺03] SCHEELE, NICOLAI, MARTIN MAUVE, WOLFGANG EFFELSBERG, ANJA WESSELS, HOLGER HORZ und STEPHAN FRIES: *The Interactive Lecture. A New Teaching Paradigm Based on Ubiquitous Computing*. In: *Poster Proceedings of the CSCL 2003*, Seiten 135–137, Bergen, Norwegen, Juni 2003. CSCL 2003.
- [Sou03] SOURCEFORGE: <http://www.sourceforge.net>. WWW, August 2003. Letzter Besuch: 25.08.2004.
- [Ste00] STEINMETZ, RALF: *Multimediatechnik*. Springer, 2000.
- [Sup90] SUPPES, P.: *Three current tutoring systems and future needs*. In: FRASSON, C. und G. GAUTHIER (Herausgeber): *Intelligent Tutoring Systems: At the crossroads of artificial intelligence and education*, Seiten 251–265. Ablex Publishing Corporation, Norwood, NJ, 1990.
- [Sve97] SVEIBY, K.E.: *The new organizational wealth: Managing & measuring knowledge based assets*. San Francisco, 1997.
- [Tea04] TEAM, OPENACS: <http://openacs.org>. WWW, August 2004. Letzter Besuch: 23.08.2004.
- [Tel98] TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU: *Packet based multimedia communication systems. Recommendation H.323*, Februar 1998. Genf, Schweiz.
- [UM04] UNIVERSITÄT MANNHEIM, LEHRSTUHL FÜR PRAKTISCHE INFORMATIK IV: http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html. WWW, 2004.
- [Wai90] WAINER, HOWARD: *Computerized Adaptive Testing: A Primer*. Lawrence Erlbaum Associates Inc., 1. Auflage, 1990.

- [web02] WEBCT INC.: *http://www.webct.com*, 2002. Letzter Besuch: 04.12.2002.
- [Wei02a] WEIBLER, JÜRGEN: *Organisation IV: Organisationaler Wandel und Organisationales Lernen*, 2002. Kursunterlagen der FernUniversität Hagen, Kurs Nr. 41053.
- [Wei02b] WEIBLER, JÜRGEN: *Unternehmensführung I: Herausforderungen moderner Unternehmensführung*, 2002. Kursunterlagen der FernUniversität Hagen, Kurs Nr. 508.
- [Wir03] WIRGES, NORBERT: *Java-Applets zur Simulation von quasi-experimentellen Designs bezüglich der Themen "Non-Equivalent Groups Design" und "Regression Discontinuity Design"*. http://www.informatik.uni-mannheim.de/pi4/info/animation/main_animation.html, Juli 2003. Studienarbeit.
- [WK04] WINSCHHEL, LILLI und STEPHAN KOPF: *Entwicklung einer Börsensimulation der multiagentenbasierten Entwicklungsumgebung NetLogo*. Technischer Bericht, Fakultät für Mathematik und Informatik; Universität Mannheim, Mannheim, Oktober 2004.
- [WM03] WEINBERGER, ARMIN und HEINZ MANDL: *Computer-mediated knowledge communication*. Technischer Bericht, Institute for Empirical Pedagogy and Pedagogical Psychology, 2003.
- [WP01] WESSNER, MARTIN und HANS-RÜDIGER PFISTER: *Group formation in computer-supported collaborative learning*. In: *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*, Seiten 24–31. ACM Press, 2001.
- [Zan76] ZANGENMEISTER, CHRISTOF: *Nutzwertanalyse in der Systemtechnik*. Wittemannsche Buchhandlung, 1976.
- [Zel00] ZELLER, ANDREAS: *Making students read and review code*. SIGCSE Bull., 32(3):89–92, 2000.
- [Zim92] ZIMBARDO, P.: *Psychologie*. Berlin, 5. Auflage, 1992.

Index

- .ltn, 53
- Überlast-Kontrolle, 140
- Übungen, 20
 - in der Fernlehre, 25
 - in der Präsenzlehre, 24
- .LRN, 128
 - Integration in CATS, 104
- 3-Parameter Logistic Model, 34
- 3PLM, 34
- Abgrenzungsfaktor, 34
- ACT-Modell, 16
- Adaptivität, 73
- ADL, 47
- Agenten-Technologie, 39
- Akzeptanz-Test, 158
- Analogie, 17
- Animation
 - interaktive, 32
- Animationen, 31
- Architektur des CATS-System, 69
- Aristoteles, 6
- Audio-/Video-Konferenz-Systeme, 56
- Aufgabenstellung
 - Anforderung an, 23
- Aufgabensysteme
 - automatisierte, 33
- Austauschnorm für Übungsaufgaben, 45
- Auswahlverfahren, 83
- Authentifizierungsdaten, 72
- Autonomie, 18
- Autorensysteme
 - zur Erstellung von Übungsaufgaben, 44
- AutoTutor, 39
- Behaviorismus, 11
 - klassische Konditionierung, 11
 - operantes Konditionieren, 12
- Betreuung
 - mentorielle, 27
- Beurteilungssysteme, 39
- Bewertung der Gruppenleistung, 91
- Blackboard Learning System, 51
 - Integration in CATS, 104
- Blended Learning, 155
- Blockveranstaltungen
 - Klausurvorbereitung, 28
- Brainstorming, 17
- Brunner, 13
- Building-Blocks-Program, 51

Index

- CACSD, 40
- Call Processing Language, 78
- CAT s. Computerunterstütztes Adaptive Testen (CAT), 34
- CATS
 - Übungsbereich, 100
 - Administration, 125
 - Anbindung an Lernplattformen, 127
 - Anmelde-Server, 110
 - Anwendungsschicht, 101, 110
 - Appletverwaltung, 125
 - Aufgabeneditor, 96
 - Aufgabenmodule, 117
 - Autorenwerkzeuge, 93, 121
 - Awareness-Funktion, 123
 - Benutzerbereich, 126
 - Benutzerinterface, 117
 - Benutzerverwaltung, 126
 - Client, 101
 - Client-Implementierung, 113
 - Controlling-Funktion, 126
 - Datenbankschicht, 100
 - Applets, 100
 - Studierende, 100
 - Tabellen, 100
 - Datenbankstruktur, 107
 - Ergebnis-Server, 111
 - Gruppenaufgabenserver, 112
 - Gruppenkommunikation, 124
 - Implementierung, 107
 - Integration in Lernplattformen, 102
 - Kommunikationsanbindung, 74
 - Kommunikationsunterstützung, 121
 - Navigation, 113
 - Server, 99
 - Server-Implementierung, 107
 - Tabellenstruktur, 108
 - Verwaltungssystem, 112
- CATS-Architektur, 69
- CATS-Aufgabeneditor, 94
- CATS-Integrationsmodell, 127
- CBAT-2, 35
- CD, 148
- Chaos
 - kreatives, 18
- Chat-Räume, 28
- ClassRoom, 42
- Client-Server-Architektur, 98
- CLIX-Campus, 50
- Cognitive apprenticeship, 13
- Cognitive Tools, 13
- Cognitive Tutor, 45
- Community, 10
- Computer Aided Control System Design, 40
- Computerunterstütztes Adaptive Testen (CAT), 34
- Congestion Control, 140
- Content Balanced Adaptive Testing, 35
- CPL, 78
- CPM, 95
- Critical Path Method, 95
- CROCODILE, 44
- CSCW, 42
- CTAT, 45
- CTS, 136
- DCT, 32
- Deduktion, 17
- Descartes, 6
- DFWMAC, 136

- Dialektik, 7
- Dialog, 16
- Difficulty Parameter, 34
- DIFS, 137
- Dijkstra-Algorithmus, 137
- Discrimination Parameter, 34
- Diskrete Kosinustransformation, 146
- Diskussion, 40
- dotLRN, 53
 - Funktionalitäten, 53
 - Implementierung, 55
- Drill & Practice, 13
- DVD, 148
- Dynamisches Modell, 16
- EACS, 112
- Einordnung von Aufgabeneditoren, 46
- Empirismus, 6
- Eon, 37
 - Document Browser, 37
 - Interaction Editor, 37
 - Meta Strategy Editor, 38
 - Presentation Contents Browser, 37
 - Student Model Editor, 38
 - Topic Contents Browser, 37
 - Topic Network Editor, 37
 - Tutoring Strategy Editor, 37
- Epistemologie, 6
- Erfahrungsaustausch, 16
- Ergebnisse, 159
- Erkenntnis
 - Definition, 5
- Evaluiierung, 155
- Evaluiierung von CATS, 131
- EXCHECK, 38
- Exercise Administration and Controlling System, 112
- Existenzialismus, 8
- Expertfinder, 44
- Exposed Terminal Problem, 135
- Externalisierung, 17
- Fehlerkorrekturverfahren, 148
- Fernlehrszenario, 25
- Flooding, 140
- Fluktuation, 18
- Fouriertransformation, 144
- G.700, 76
- G.728, 76
- Gatekeeper, 76
- General Systems Performance Theory, 41
- Gerichtsverhandlung
 - simulierte, 93
- Gruppenübung, 92
- Gruppenarbeit, 22
- Gruppenaufgabe, 92
- Gruppenbewertung
 - absolute, 91
 - relative, 91
- Gruppenverwaltung
 - Tabelle, 100
- Gruppenziel
 - Definition des, 22
- GSPT, 41
- H.225, 76
- H.245, 76
- H.261, 76
- H.263, 76
- H.264, 76

Index

- H.323, 75, 123
 - Grundlagen, 75
- Höhlengleichnis, 6
- Heckhausens Modell, 23
- Hegel, 7
- Heidegger, 8
- Hidden Terminal Problem, 134
- HRG, 28
- Huffman-Kodierung, 143
- Hypothesen, 156
- ICQ, 57
- Idealismus
 - transzendentaler, 7
- imc, 50
- IMS QTIS, 45
- Individualbewertung, absolut, 80
- Induktion, 17
- Innovation, 19
- Instant-Messenger, 56
- Integrationsmodell, 103
- Intelligente Tutorielle Systeme, 35
- Intention, 18
- Interaktion
 - soziale, 27
- Interaktionsfeld, 19
- Interaktivität
 - soziale, 39
- Internalisierung, 18
- Item Response Theory, 34
- Item-Response-Theorie, 70
- ITR, 34
- ITS, 35
 - Aufbau von, 35
 - Autorensystem, 37
 - Beispiel-Anwendungen, 36
 - Expertenmodul, 35
 - Kommunikationsmodul, 36
 - Kritik, 38
 - Studentenmodul, 36
 - Unterrichtsmodul, 36
- ITU, 75
- JaTeK, 42
- Java Based Workgroup Support, 42
- JaWoS, 42
- Kant, 7
- Kartesianischer Dualismus, 7
- Kennzahlensystem, 70
- Knowledge Communities, 14
- kognitive Strukturen, 13
- Kognitivismus, 13
- Kombination, 18
- Kommunikation
 - Beschränkung, 26
- Kommunikationssysteme, 55
- Konditionierung
 - klassische, 11
 - operante, 12
- Konstruktion, 15
- Konstruktivismus, 13, 39
 - sozialer, 14, 92
- Kontaktlisten, 28
- Konzeptualisierung, 15
- KPI, 118
- Learning by doing, 19
- Learning-Cycle, 14
- Lehre
 - in die Lehre gehen, 17
- Lehrleistung, 91
- Leistungsbewertung, 91

- Leitungskodierung, 132
- Lern- und Übungsumgebungen, 47
- Lernen, 10
 - Definition, 5, 11
 - kooperatives, 22
- Lernfortschritt, 80
- Lerngruppen
 - Systeme zur Unterstützung von, 43
- Lernmaschinen, 13
- Lernplattformen
 - Integration, 127
 - Integrationsmöglichkeiten, 104
- Lernraum Virtuelle Universität, 52
- Lerntheorien, 11
- Lernwirksamkeit, 32
- Lernziele, 28
- LMS, 49
- Locke, 6
- LOM, 49
- LVU, 52

- Marx, 7
- Mbone-Tools, 88
- MCU, 89, 124
- Medientransport, 76
- Meinungsführer, 23
- Meta-Wissen, 10
- Metapher, 17
- Metra Potential Method, 95
- Modelle, 17
- Moot-Court, 93, 153
- Motivationsmodelle
 - für Gruppenarbeit, 23
- MPM, 95
- MSI, 40
- MSN-Messenger, 57

- Multicast Routing, 138
- Multiple-Choice
 - Editor, 93
- Multipoint Control Unit, 89

- Netmeeting, 75
- Netzplantechnik, 95, 96
- News-Foren, 28
- Non-Equivalent-Group-Design, 152
- Nonaka, 16
- Nutzwertmethode, 72

- Objektive Tatbestandsmerkmale, 154
- Objektivismus, 13
- Objektzentrierte Kommunikation, 116
- Offene Lösung, 23
- Open Loop, 41
- Open Masterplan, 23
- OpenACS, 55
- OpenMCU, 89
- Organismus, 12

- Parameterverhandlung, 76
- Pawlowsche Hunde, 11
- Pawlowski, 11
- Peer Assessment, 41
- Peer-to-Peer-Kommunikation, 74, 122
- Performance Indicator, 72
- Performance, 72, 73
- PERT, 96
- Phänomologie, 8
- Physik-Übungssystem, 33
- Piaget, 13
- PIFS, 137
- PIM Sparse-Mode, 140
- Plato, 6
- Platten-Scheduling, 147

Index

- Politikon, 156
- Prädikatenlogik, 35
- Präsenzübungen, 27
- Praktomat, 43
- Problemstellung, 2
- Proficiency, 70
- Program Evaluation and Review, 96
- Programmieraufgaben, 43
- Prolog, 36
- Pseudo-Guessing Paramater, 34
- Q.931, 76
- Rückkopplung, 12
- Randomized Response Technik, 151
- Rangliste, 73
- RAS, 76
- Ratefaktor, 34
- Rationalismus, 5
 - Kontinentaler, 6
- Real-Time Protocol Control Protocol, 76
- Realtime Protocol, 76
- Rechtswidrigkeit, 154
- Redundanz, 19
- Reed-Salomon-Product-Code, 149
- Reflexion
 - kollektive, 17
- Registration and Admission Status, 76
- Registrierung
 - der Teilnehmer, 76
- Reiz
 - konditionierter, 12
 - unkonditionierter, 12
- Reiz-Reaktions-Verbindung, 12
- Reliability, 71
- Response-Codes, 78
- Ressourceninterdependenz, 22, 23
- Results, 100
- rfc2543, 77
- RRT, 151
- RTCP, 76
- RTP, 76
- RTS, 136
- S-R-Verbindungen, 11
- Schuld, 154
- Schwierigkeitsgrad, 34, 70
- SCO, 49
- SCORM, 47, 51
 - Laufzeitumgebung, 50
 - Metadaten, 49
- Secret Masterplan, 23
- Selbstbeurteilung, 40
- Self-Monitoring, 80
- Sequenzierung, 23
- Session Initiation Protocol, 77
- Shared Whiteboard, 76
- Shortest Path-Algorithmus, 137
- SIFS, 137
- Simulation, 40
- Simulationen, 33
- SIP, 77, 124
 - ACK, 78
 - BYE, 78
 - CANCEL, 78
 - Infrastruktur-Elemente, 79
 - Invite, 78
 - Nachrichten-Definition, 78
 - OPTIONS, 78
 - Protokoll, 77
 - Proxy Server, 79
 - REGISTER, 78

- Registrar Server, 79
- User Agent, 79
- SIP Redirect Server, 79
- Skinner, 11, 12
- Skripting, 23
- Sozialisation, 16
- Sprechstunden, 27
- Stimulus, 12
- Strafbarkeitsmerkmale, 154
- Subjektive Tatbestandsmerkmale, 154
- Systemadministration, 125
- Szenario
 - Offline-Übung, 74, 81
 - Online-Übung mit Kommunikationsunterstützung, 81
 - Unterstützung von Gruppenübungen, 82
- Szenario 3, 108
- T.120, 76
- Talking Heads, 39
- TCP, 140
- Teambildung, 83
- Token Ring, 134
- Transmission Control Protocol, 140
- Truncated Reverse PAth Broadcasting, 140
- ULI, 155
- Usability-Design, 113
- VALID, 38
- Variantengenerator, 43
- Verantwortlichkeit
 - individuelle, 22
- Verbindungsaufbau, 76
- Verhalten
 - operantes, 12
 - respondentes, 12
- Verstärkung, 12
- Vielfalt, 19
- Virtuelle Universität, 52
- Virtuelles Rennen, 40
- VITAL, 43
- VU, 52
- Wainer, 70
- WebAssign, 52
- WebCall, 42
- WebCT, 51
- WI-MA-Editor, 45
- Winfoline, 50, 155
- Wireless Local Area Network, 134
- Wissen
 - Definition, 5, 9
 - erster Ordnung, 10
 - explizites, 10
 - implizites, 10
 - zweiter Ordnung, 10
- Wissensspirale, 16
 - Voraussetzungen, 18
- Wissensumwandlung, 16
- Wissensvermittlung, 1
- WLAN, 134
- Zeit-Divergenz, 25
- Zeitbedarf, 71
- Zeugnisergänzung, 91
- Zuverlässigkeit, 71